

# Better Guarantees for $k$ -Means and Euclidean $k$ -Median by Primal-Dual Algorithms\*

Sara Ahmadian<sup>†</sup>    Ashkan Norouzi-Fard<sup>‡</sup>    Ola Svensson<sup>§</sup>    Justin Ward<sup>¶</sup>

December 26, 2016

## Abstract

Clustering is a classic topic in optimization with  $k$ -means being one of the most fundamental such problems. In the absence of any restrictions on the input, the best known algorithm for  $k$ -means with a provable guarantee is a simple local search heuristic yielding an approximation guarantee of  $9 + \epsilon$ , a ratio that is known to be tight with respect to such methods.

We overcome this barrier by presenting a new primal-dual approach that allows us to (1) exploit the geometric structure of  $k$ -means and (2) to satisfy the hard constraint that at most  $k$  clusters are selected without deteriorating the approximation guarantee. Our main result is a 6.357-approximation algorithm with respect to the standard LP relaxation. Our techniques are quite general and we also show improved guarantees for the general version of  $k$ -means where the underlying metric is not required to be Euclidean and for  $k$ -median in Euclidean metrics.

---

\*Supported by ERC Starting Grant 335288-OptApprox.

<sup>†</sup>Department of Combinatorics and Optimization, University of Waterloo. Email: [sahmadian@uwaterloo.ca](mailto:sahmadian@uwaterloo.ca).

<sup>‡</sup>School of Computer and Communication Sciences, EPFL. Email: [ashkan.norouzifard@epfl.ch](mailto:ashkan.norouzifard@epfl.ch).

<sup>§</sup>School of Computer and Communication Sciences, EPFL. Email: [ola.svensson@epfl.ch](mailto:ola.svensson@epfl.ch).

<sup>¶</sup>School of Computer and Communication Sciences, EPFL. Email: [justin.ward@epfl.ch](mailto:justin.ward@epfl.ch).

# 1 Introduction

Clustering problems have been extensively studied in computer science. They play a central role in many areas, including data mining and knowledge discovery, data compression and vector quantization, pattern recognition and classification, and detection of abnormal data. Perhaps the most widely considered clustering problem is the so-called *k-means* problem: given a set  $\mathcal{D}$  of  $n$  points in  $\mathbb{R}^\ell$  and an integer  $k$ , the task is to select a set  $S$  of  $k$  *cluster centers* in  $\mathbb{R}^\ell$ , so that  $\sum_{j \in \mathcal{D}} c(j, S)$  is minimized, where  $c(j, S)$  is the squared Euclidean distance between  $j$  and its nearest center in  $S$ .

The *k-means* problem has been well-studied experimentally and practically [15]. One of the most commonly used heuristics for *k-means* is Lloyd’s algorithm [24], which is based on iterative improvement. Despite its ubiquity in practice, Lloyd’s algorithm has, in general, no worst-case guarantee and may not even converge in polynomial time [2, 28]. Arthur and Vassilvitskii [3] proposed a randomized initialization procedure for Lloyd’s algorithm, called *k-means++*, that leads to a  $\Theta(\log k)$  expected approximation guarantee in the worst case. Under additional assumptions about the *clusterability* of the input dataset, Ostrovsky et al. [26] showed that Lloyd’s algorithm gives a PTAS for *k-means* clustering.

Various other PTASes have been developed for restricted instances of the *k-means* problem under a variety of assumptions. For example, Awasthi, Blum, and Sheffet obtain a PTAS assuming the instance has certain stability properties [5], and there has been a long line of work (beginning with [25]) obtaining better and better PTASes under the assumption that  $k$  is constant. Most recently, it has been shown that local search gives a PTAS under the assumption that the dimension  $\ell$  of the dataset is constant [11, 13]. These last results generalize to the case in which the squared distances are from the shortest path metric on a graph with forbidden minors [11] or from a metric with constant doubling dimension [13].

Under no additional assumptions, however, the best approximation for the general *k-means* problem has for some time remained a  $(9 + \epsilon)$ -approximation algorithm based on local search, due to Kanungo et al. [19]. Moreover, their analysis shows that no natural local search algorithm performing a fixed number of swaps can improve upon this ratio. In terms of hardness, Awasthi, Charikar, Krishnaswamy, and Sinop [6] showed that *k-means* is APX-hard, and so we cannot hope for a PTAS in the general case. Follow-up work by Lee, Schmidt, and Wright [20] shows that it is NP-hard to approximate this problem to within a factor better than 1.0013.

In summary, while *k-means* is perhaps the most widely used clustering problem in computer science, the only constant-factor approximation algorithm for the general case is based on simple local search heuristics. This is in stark contrast to many other well-studied clustering problems, such as facility location and *k-median*. Over the past several decades, work on these problems has been responsible for the refinement of a variety of core techniques in approximation algorithms such as dual fitting, primal-dual, and LP-rounding [27, 10, 7, 21, 23, 18, 17, 22, 16]. The development of these techniques has led to several breakthroughs giving the current best approximation guarantees for both facility location (a 1.488-approximation due to Li [21]) and *k-median* (a 2.675-approximation due to Byrka et al. [8]). In both cases, LP-based techniques now give significantly better results than previous local search algorithms [4, 9]. These techniques have not yet been able to attain similar improvements for *k-means* primarily because they have relied heavily on the triangle inequality, which does not hold in the case of *k-means*.

**Our results** In this work, we overcome this barrier by developing new techniques that allow us to exploit the standard LP formulation for *k-means*. We significantly narrow the gap between known upper and lower bounds by designing a new primal-dual algorithm for the *k-means* problem. We stress that our algorithm works in the general case that  $k$  and  $\ell$  are part of the input, and requires

no additional assumptions on the dataset.

**Theorem 1.1.** *For any  $\epsilon > 0$ , there is a  $(\rho_{\text{mean}} + \epsilon)$ -approximation algorithm for the  $k$ -means problem, where  $\rho_{\text{mean}} \approx 6.357$ . Moreover, the integrality gap of the standard LP is at most  $\rho_{\text{mean}}$ .*

We now describe our approach and contributions at a high level. Given a  $k$ -means instance, we apply standard discretization techniques (e.g., [12]) to obtain an instance of the *discrete  $k$ -means* problem, in which we are given a discrete set  $\mathcal{F}$  of candidate centers in  $\mathbb{R}^\ell$  and must select  $k$  centers from  $\mathcal{F}$ , rather than  $k$  arbitrary points in  $\mathbb{R}^\ell$ . This step incurs an arbitrarily small loss in the approximation guarantee with respect to the original  $k$ -means instance. Using Lagrangian relaxation, we can then consider the resulting discrete problem using the standard linear programming formulation for facility location.

Our approach then starts with the framework of Jain and Vazirani [18] for the  $k$ -median problem. In their paper, they first present a *Lagrangian Multiplier Preserving* (LMP) 3-approximation algorithm for the facility location problem. Then they run binary search over the opening cost of the facilities, and use the aforementioned algorithm to get two solutions: one that opens more than  $k$  facilities and one that opens less than  $k$ , such that the opening cost of facilities in these solutions are close to each other. These solutions are then combined to obtain a solution that opens exactly  $k$  facilities. This step results in losing another factor 2 in the approximation guarantee, which results in a 6-approximation algorithm for  $k$ -median. The factor 6 was later improved by Jain, Mahdian, and Saberi [17] who obtained a 4-approximation algorithm for  $k$ -median by developing an LMP 2-approximation algorithm for facility location.

One can see that the same approach gives a much larger constant factor for the  $k$ -means problem since one cannot anymore rely on the triangle inequality. We use two main ideas to overcome this obstacle: (1) we exploit the geometric structure of  $k$ -means and (2) we develop a new primal-dual approach. Specifically, we modify the primal-dual algorithm of Jain and Vazirani [18] into a parameterized version which allows us to regulate the “aggressiveness” of the opening strategy of facilities. By using properties of Euclidean metrics we show that this leads to improved LMP approximation algorithms for  $k$ -means.

By the virtue of [1], these results already imply upper bounds on the integrality gaps of the standard LP relaxations, albeit with a rounding algorithm that may require exponential time. Our second main contribution is a new polynomial time algorithm that accomplishes the same task. Several new ideas are required to obtain this rounding. Specifically, instead of finding two solutions by binary search as in the framework of [18], we find a sequence of solutions such that the opening costs and also the dual values of any two consecutive solutions are close in  $L^\infty$ -norm. We show how to combine two appropriate such solutions to obtain a solution that opens exactly  $k$  facilities while losing only a factor  $1 + \epsilon$  in the approximation guarantee.

**Extensions to other problems** In addition to the standard  $k$ -means problem, we show that our results also extend to the following two problems. In the first extension, we consider the Euclidean  $k$ -median problem. Here we are given a set  $\mathcal{D}$  of  $n$  points in  $\mathbb{R}^\ell$  and a set  $\mathcal{F}$  of  $m$  points in  $\mathbb{R}^\ell$  corresponding to facilities. The task is to select a set  $S$  of at most  $k$  facilities from  $\mathcal{F}$  so as to minimize  $\sum_{j \in \mathcal{D}} c(j, S)$ , where  $c(j, S)$  is now the (non-squared) Euclidean distance from  $j$  to its nearest facility in  $S$ . For this problem, no approximation better than the general 2.675-approximation algorithm of Byrka et al. [8] for  $k$ -median was known.

**Theorem 1.2.** *For any  $\epsilon > 0$ , there is a  $(\rho_{\text{med}} + \epsilon)$ -approximation algorithm for the Euclidean  $k$ -median problem, where  $\rho_{\text{med}} \approx 2.633$ . Moreover, the integrality gap of the standard LP is at most  $\rho_{\text{med}}$ .*

In the second extension, we consider a variant of the  $k$ -means problem in which each  $c(j, S)$  corresponds to the squared distance in an arbitrary (possibly non-Euclidean) metric on  $\mathcal{D} \cup \mathcal{F}$ . For this problem, the best-known approximation algorithm is a 16-approximation due to Gupta and Tangwongsan [14]. In this paper, we obtain the following improvement:

**Theorem 1.3.** *For any  $\epsilon > 0$ , there is a  $(9 + \epsilon)$ -approximation algorithm for the  $k$ -means problem in general metrics. Moreover, the integrality gap of the standard LP is at most 9.*

We remark that the same hardness reduction as used for  $k$ -median [17] immediately yields a much stronger hardness result for the above generalization than what is known for the standard  $k$ -means problem: it is hard to approximate the  $k$ -means problem in general metrics within a factor  $1 + 8/e - \epsilon \approx 3.94$  for any  $\epsilon > 0$ .

**Open questions.** We present a fairly clean quasi-polynomial time algorithm and we show that these techniques can be generalized to achieve a polynomial running time in a rather complex way. It is a nice open problem to get a simpler algorithm with polynomial running time.

Additionally, we note that while Jain, Mahdian, Saberi [17] present an LMP 2-approximation algorithm for the facility location problem, we are not yet able to use their algorithm for our approach (and thereby improve the approximation guarantee for  $k$ -median). This is because it is unclear how to obtain a sequence of *close* dual solutions by using those techniques and therefore we do not know how to guarantee that the algorithm will open exactly  $k$  facilities by only losing a factor  $(1 + \epsilon)$  in that case. An interesting open question is whether it is possible to combine their techniques with ours to give a 2-approximation for  $k$ -median.

**Outline of paper.** In Section 2 we review the standard LP formulation that we use, as well as its Lagrangian relaxation. We then in Section 3 show how to exploit the geometric structure  $k$ -means and Euclidean  $k$ -median to give improved LMP guarantees. In Section 4 we show the main ideas behind our new rounding approach by giving an algorithm that runs in quasi-polynomial time. These results are then generalized in Sections 5, 6, and 7 to obtain an algorithm that runs in polynomial time.

## 2 The standard LP relaxation and its Lagrangian relaxation

Here and in the remainder of the paper, we shall consider the *discrete*  $k$ -means problem, where we are given a discrete set  $\mathcal{F}$  of facilities (corresponding to candidate centers).<sup>1</sup> Henceforth, we will simply refer to the discrete  $k$ -means problem as the  $k$ -means problem.

Given an instance  $(\mathcal{D}, \mathcal{F}, d, k)$  of the  $k$ -means problem or the  $k$ -median problem, let  $c(j, i)$  denote the connection cost of client  $j$  if connected to facility  $i$ . That is,  $c(j, i) = d(j, i)$  in the case of  $k$ -median and  $c(j, i) = d(j, i)^2$  in the case of  $k$ -means. Let  $n = |\mathcal{D}|$  and  $m = |\mathcal{F}|$ .

The standard linear programming (LP) relaxation of these problems has two sets of variables: a variable  $y_i$  for each facility  $i \in \mathcal{F}$  and a variable  $x_{ij}$  for each facility-client pair  $i \in \mathcal{F}, j \in \mathcal{D}$ . The intuition of these variables is that  $y_i$  should indicate whether facility  $i$  is opened and  $x_{ij}$  should indicate whether client  $j$  is connected to facility  $i$ . The standard LP relaxation can now be formulated as follows.

---

<sup>1</sup>As discussed in the introduction, it is well-known that a  $\rho$ -approximation algorithm for this case can be turned into a  $(\rho + \epsilon)$ -approximation algorithm for the standard  $k$ -means problem, for any constant  $\epsilon > 0$  (see e.g., [12]).

$$\begin{aligned} \min \quad & \sum_{i \in \mathcal{F}, j \in \mathcal{D}} x_{ij} \cdot c(j, i) \\ \text{s.t.} \quad & \sum_{i \in \mathcal{F}} x_{ij} \geq 1 \quad \forall j \in \mathcal{D} \end{aligned} \quad (2.1)$$

$$x_{ij} \leq y_i \quad \forall j \in \mathcal{D}, i \in \mathcal{F} \quad (2.2)$$

$$\sum_{i \in \mathcal{F}} y_i \leq k \quad (2.3)$$

$$x, y \geq 0. \quad (2.4)$$

The first set of constraints says that each client should be connected to at least one facility; the second set of constraints enforces that clients can only be connected to opened facilities; and the third constraint says that at most  $k$  facilities can be opened. We remark that this is a relaxation of the original problem as we have relaxed the constraint that  $x$  and  $y$  should take Boolean values to a non-negativity constraint. For future reference, we let  $\text{OPT}_k$  denote the value of an optimal solution to this relaxation.

A main difficulty for approximating the  $k$ -median and the  $k$ -means problems is the hard constraint that at most  $k$  facilities can be selected, i.e., constraint (2.3) in the above relaxation. A popular way of overcoming this difficulty, pioneered in this context by Jain and Vazirani [18], is to consider the Lagrangian relaxation where we multiply the constraint (2.3) times a Lagrange multiplier  $\lambda$  and move it to the objective. This results, for every  $\lambda \geq 0$ , in the following relaxation and its dual that we denote by  $\text{LP}(\lambda)$  and  $\text{DUAL}(\lambda)$ , respectively.

$\text{LP}(\lambda)$	$\text{DUAL}(\lambda)$
$\begin{aligned} \min \quad & \sum_{i \in \mathcal{F}, j \in \mathcal{D}} x_{ij} \cdot c(j, i) + \lambda \cdot \left( \sum_{i \in \mathcal{F}} y_i - k \right) \\ \text{s.t.} \quad & (2.1), (2.2), \text{ and } (2.4). \end{aligned}$	$\begin{aligned} \max \quad & \sum_{j \in \mathcal{D}} \alpha_j - \lambda \cdot k \\ \text{s.t.} \quad & \sum_{j \in \mathcal{D}} [\alpha_j - c(j, i)]^+ \leq \lambda \quad \forall i \in \mathcal{F} \quad (2.5) \\ & \alpha \geq 0. \end{aligned}$

Here, we have simplified the dual by noticing that the dual variables  $\{\beta_{ij}\}_{i \in \mathcal{F}, j \in \mathcal{D}}$  corresponding to the constraints (2.2) of the primal can always be set  $\beta_{ij} = [\alpha_j - c(j, i)]^+$ ; the notation  $[a]^+$  denotes  $\max(a, 0)$ . Moreover, to see that  $\text{LP}(\lambda)$  remains a relaxation, note that any feasible solution to the original LP is a feasible solution to the Lagrangian relaxation of no higher cost. In other words, for any  $\lambda \geq 0$ , the optimum value of  $\text{LP}(\lambda)$  is at most  $\text{OPT}_k$ .

If we disregard the constant term  $\lambda \cdot k$  in the objective functions,  $\text{LP}(\lambda)$  and  $\text{DUAL}(\lambda)$  become the standard LP formulation and its dual for the facility location problem where the opening cost of each facility equals  $\lambda$  and the connection costs are defined by  $c(\cdot, \cdot)$ . Recall that the facility location problem (with uniform opening costs) is defined as the problem of selecting a set  $S \subseteq \mathcal{F}$  of facilities to open so as to minimize the opening cost  $|S|\lambda$  plus the connection cost  $\sum_{j \in \mathcal{D}} c(j, S)$ . Jain and Vazirani [18] introduced the following method for addressing the  $k$ -median problem motivated by simple economics. On the one hand, if  $\lambda$  is selected to be very small, i.e., it is cheap to open facilities, then a good algorithm for the facility location problem will open many facilities. On the other hand, if  $\lambda$  is selected to be very large, then a good algorithm for the facility location problem will open few facilities. Ideally, we want to use this intuition to find an opening price that leads to the opening of exactly  $k$  facilities and thus a solution to the original, constrained problem.

To make this intuition work, we need the notion of *Lagrangian Multiplier Preserving* (LMP) approximations: we say that a  $\rho$ -approximation algorithm is LMP for the facility location problem with opening costs  $\lambda$  if it returns a solution  $S \subseteq \mathcal{F}$  satisfying

$$\sum_{j \in \mathcal{D}} c(j, S) \leq \rho(\text{OPT}(\lambda) - |S|\lambda),$$

where  $\text{OPT}(\lambda)$  denotes the value of an optimal solution to  $\text{LP}(\lambda)$  without the constant term  $\lambda \cdot k$ . The importance of this definition becomes apparent when either  $\lambda = 0, |S| \leq k$  or  $|S| = k$ . In those cases, we can see that the value of the  $k$ -median or  $k$ -means solution is at most  $\rho$  times the optimal value of its relaxation  $\text{LP}(\lambda)$ , and thus an  $\rho$ -approximation with respect to its standard LP relaxation since  $\text{OPT}(\lambda) - k \cdot \lambda \leq \text{OPT}_k$  for any  $\lambda \geq 0$ .

### 3 Exploiting Euclidean metrics via primal-dual algorithms

In this section we show how to exploit the structure of Euclidean metrics to achieve better approximation guarantees. Our algorithm is based on the primal-dual algorithm for the facility location problem by Jain and Vazirani [18]. We refer to their algorithm as the JV algorithm. The main modification to their algorithm is that we allow for a more “aggressive” opening strategy of facilities. The amount of aggressiveness is measured by the parameter  $\delta$ : we devise an algorithm  $\text{JV}(\delta)$  for each parameter  $\delta \geq 0$ , where a smaller  $\delta$  results in a more aggressive opening strategy. We first describe  $\text{JV}(\delta)$  and we then optimize  $\delta$  for the considered objectives to obtain the claimed approximation guarantees.

We remark that the result in [1] (non-constructively) upper bounds the integrality gap of the standard LP relaxation of  $k$ -median in terms of the LMP approximation guarantee of JV. This readily generalizes to the  $k$ -means problem and  $\text{JV}(\delta)$ . Consequently, our guarantees presented here upper bound the integrality gaps as the theorems state in the introduction.

#### 3.1 Description of $\text{JV}(\delta)$

As alluded to above, the algorithm is a modification of JV, and Remark 3.2 below highlights the only difference. The algorithm consists of two phases: the dual-growth phase and the pruning phase.

**Dual-growth phase:** In this stage, we construct a feasible dual solution  $\alpha$  to  $\text{DUAL}(\lambda)$ . Initially, we set  $\alpha = \mathbf{0}$  and let  $A = \mathcal{D}$  denote the set of active clients (which is all clients at first). We then repeat the following until there are no active clients, i.e.,  $A = \emptyset$ : increase the dual-variables  $\{\alpha_j\}_{j \in A}$  corresponding to the active clients at a uniform rate until one of the following events occur (if several events happen at the same time, break ties arbitrarily):

Event 1: A dual constraint  $\sum_{j \in \mathcal{D}} [\alpha_j - c(j, i)]^+ \leq \lambda$  becomes tight for a facility  $i \in \mathcal{F}$ . In this case we say that facility  $i$  is *tight* or *temporarily opened*. We update  $A$  by removing the active clients with a *tight edge* to  $i$ , that is, a client  $j \in A$  is removed if  $\alpha_j - c(j, i) \geq 0$ . For future reference, we say that facility  $i$  is the *witness* of these removed clients.

Event 2: An active client  $j \in A$  gets a tight edge, i.e.,  $\alpha_j - c(j, i) = 0$ , to some already tight facility  $i$ . In this case, we remove  $j$  from  $A$  and let  $i$  be its witness.

This completes the description of the dual-growth phase. Before proceeding to the pruning phase, let us remark that the constructed  $\alpha$  is indeed a feasible solution to  $\text{DUAL}(\lambda)$  by design. It is

clear that  $\alpha$  is non-negative. Now consider a facility  $i \in \mathcal{F}$  and its corresponding dual constraint  $\sum_{j \in \mathcal{D}} [\alpha_j - c(j, i)]^+ \leq \lambda$ . On the one hand, the constraint is clearly satisfied if it never becomes tight during the dual-growth phase. On other hand, if it becomes tight, then all clients with a tight edge to it are removed from the active set of clients by Event 1. Moreover, if any client gets a tight edge to  $i$  in subsequent iterations it gets immediately removed from the set of active clients by Event 2. Therefore the left-hand-side of the constraint will never increase (nor decrease) after it becomes tight so the constraint remains satisfied. Having proved that  $\alpha$  is a feasible solution to  $\text{DUAL}(\lambda)$ , let us now describe the pruning phase.

**Pruning phase:** After the dual-growth phase (too) many facilities are temporarily opened. The pruning phase will select a subset of these facilities to open. In order to formally describe this process, we need the following notation. For a client  $j$ , let  $N(j) = \{i \in \mathcal{F} : \alpha_j - c(j, i) > 0\}$  denote the facilities to which client  $j$  contributes to the opening cost. Similarly, for  $i \in \mathcal{F}$ , let  $N(i) = \{j \in \mathcal{D} : \alpha_j - c(j, i) > 0\}$  denote the clients with a positive contribution toward  $i$ 's opening cost. For a temporarily opened facility  $i$ , let

$$t_i = \max_{j \in N(i)} \alpha_j,$$

and by convention let  $t_i = 0$  if  $N(i) = \emptyset$  (this convention will be useful in future sections and will only be used when the opening cost  $\lambda$  of facilities are set to 0). Note that, if  $N(i) \neq \emptyset$ , then  $t_i$  equals the “time” that facility  $i$  was temporarily opened in the dual-growth phase. A crucial property of  $t_i$  that follows from the construction of  $\alpha$  is the following.

**Claim 3.1.** *For a client  $j$  and its witness  $i$ ,  $\alpha_j \geq t_i$ . Moreover, for any  $j' \in N(i)$  we have  $t_i \geq \alpha_{j'}$ .*

A key ingredient for the pruning phase is the *client-facility graph*  $G$  and the *conflict graph*  $H$ . The vertex set of  $G$  consist of all the clients and all facilities  $i$  such that  $\sum_{j \in \mathcal{D}} [\alpha_j - c(j, i)]^+ = \lambda$  (i.e., the tight or temporarily open facilities). There is an edge between facility  $i$  and client  $j$  if  $i \in N(j)$ . The conflict graph  $H$  is defined based on the client-facility graph  $G$  and  $t$  as follows:

- The vertex set consists of all facilities in  $G$ .
- There is an edge between two facilities  $i$  and  $i'$  if some client  $j$  is adjacent to both of them in  $G$  and  $c(i, i') \leq \delta \min(t_i, t_{i'})$ .

The pruning phase now finds a (inclusion-wise) maximal independent set  $\text{IS}$  of  $H$  and opens those facilities; clients are connected to the closest facility in  $\text{IS}$ .

**Remark 3.2.** *The only difference between the original algorithm  $\text{JV}$  and our modified  $\text{JV}(\delta)$  is the additional condition  $c(i, i') \leq \delta \min(t_i, t_{i'})$  in the definition of the conflict graph. Notice that if we select a smaller  $\delta$  we will have fewer edges in  $H$ . Therefore a maximal independent set will likely grow in size, which results in a more “aggressive” opening strategy.*

### 3.2 Analysis of $\text{JV}(\delta)$ for the considered objectives

In the following subsections, we optimize  $\delta$  and analyze the guarantees obtained by the algorithm  $\text{JV}(\delta)$  for the objective functions: k-means objective in general metrics, standard  $k$ -means objective (in Euclidean metrics), and k-median objective in Euclidean metrics. The first analysis is very similar to the original  $\text{JV}$  analysis and can also serves as a motivation for the possible improvements in Euclidean metrics.

### 3.2.1 $k$ -Means objective in general metrics

We consider the case when  $c(j, i) = d(j, i)^2$  and  $d$  forms a general metric. We let  $\delta = \infty$  so  $JV(\delta)$  becomes simply the JV algorithm. We prove the following.

**Theorem 3.3.** *Let  $d$  be any metric on  $\mathcal{D} \cup \mathcal{F}$  and suppose that  $c(j, i) = d(j, i)^2$  for every  $i \in \mathcal{F}$  and  $j \in \mathcal{D}$ . Then, for any  $\lambda \geq 0$ , Algorithm  $JV(\infty)$  constructs a solution  $\alpha$  to  $DUAL(\lambda)$  and returns a set  $\mathbf{IS}$  of opened facilities such that*

$$\sum_{j \in \mathcal{D}} c(j, \mathbf{IS}) \leq 9 \cdot \left( \sum_{j \in \mathcal{D}} \alpha_j - \lambda |\mathbf{IS}| \right).$$

*Proof.* Consider any client  $j \in \mathcal{D}$ . We shall prove that

$$\frac{c(j, \mathbf{IS})}{9} \leq \alpha_j - \sum_{i \in N(j) \cap \mathbf{IS}} (\alpha_j - c(j, i)) = \alpha_j - \sum_{i \in \mathbf{IS}} [\alpha_j - c(j, i)]^+. \quad (3.1)$$

The statement then follows by summing up over all clients and noting that any facility  $i \in \mathbf{IS}$  was temporarily opened and thus we have  $\sum_{j \in \mathcal{D}} [\alpha_j - c(j, i)]^+ = \lambda$ .

To prove (3.1), we first note that  $|\mathbf{IS} \cap N(j)| \leq 1$ . Indeed, consider  $i \neq i' \in N(j)$ . Then  $(j, i)$  and  $(j, i')$  are edges in the client-facility graph  $G$  and as  $\delta = \infty$ ,  $i$  and  $i'$  are adjacent in the conflict graph  $H$ . Hence, the temporarily opened facilities in  $N(j)$  form a clique and at most one of them can be selected in the maximal independent set  $\mathbf{IS}$ . We complete the analysis by considering the two cases  $|\mathbf{IS} \cap N(j)| = 1$  and  $|\mathbf{IS} \cap N(j)| = 0$ .

Case  $|\mathbf{IS} \cap N(j)| = 1$ : Let  $i^*$  be the unique facility in  $\mathbf{IS} \cap N(j)$ . Then

$$\frac{c(j, \mathbf{IS})}{9} \leq c(j, \mathbf{IS}) \leq c(j, i^*) = \alpha_j - (\alpha_j - c(j, i^*)) = \alpha_j - \sum_{i \in N(j) \cap \mathbf{IS}} (\alpha_j - c(j, i)).$$

Notice the amount of slack in the above analysis (specifically, the first inequality). In the Euclidean case, we exploit this slack for a more aggressive opening and to improve the approximation guarantee.

Case  $|\mathbf{IS} \cap N(j)| = 0$ : Let  $i_1$  be  $j$ 's witness. First, if  $i_1 \in \mathbf{IS}$  then by the same arguments as above we have the desired inequality; specifically, since  $j$  has a tight edge to  $i_1$  but  $i_1 \notin N(j)$  we must have  $\alpha_j = c(i_1, j)$ . Now consider the more interesting case when  $i_1 \notin \mathbf{IS}$ . As  $\mathbf{IS}$  is a maximal independent set in  $H$ , there must be a facility  $i_2 \in \mathbf{IS}$  that is adjacent to  $i_1$  in  $H$ . By definition of  $H$ , there is a client  $j_1$  such that  $(j_1, i_1)$  and  $(j_1, i_2)$  are edges in the client-facility graph  $G$ , i.e.,  $j_1 \in N(i_1) \cap N(i_2)$ . By the definition of witness and  $N(\cdot)$ , we have

$$\alpha_j \geq c(j, i_1), \quad \alpha_{j_1} > c(j_1, i_1), \quad \alpha_{j_1} > c(j_1, i_2),$$

and by the description of the algorithm (see Claim 3.1 in Section 3) we have  $\alpha_j \geq t_{i_1} \geq \alpha_{j_1}$ . Hence, using the triangle inequality and that  $(a + b + c)^2 \leq 3(a^2 + b^2 + c^2)$ ,

$$\begin{aligned} c(j, \mathbf{IS}) &\leq c(j, i_2) = d(j, i_2)^2 \leq (d(j, i_1) + d(j_1, i_1) + d(j_1, i_2))^2 \\ &\leq 3(d(j, i_1)^2 + d(j_1, i_1)^2 + d(j_1, i_2)^2) \\ &= 3(c(j, i_1) + c(j_1, i_1) + c(j_1, i_2)) \leq 9\alpha_j. \end{aligned} \quad (3.2)$$

As  $\sum_{i \in N(j) \cap \mathbf{IS}} (\alpha_j - c(j, i)) = 0$ , this completes the proof of this case and thus the theorem.  $\square$



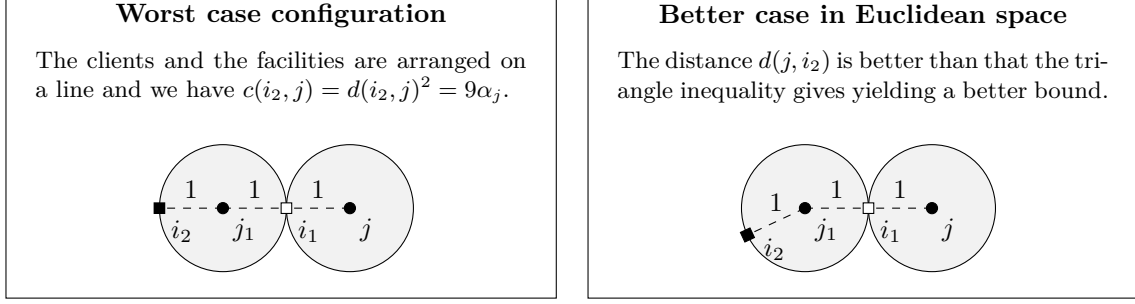


Figure 1: The intuition how we improve the guarantee in the Euclidean case. In both cases, we have  $\alpha_j = \alpha_{j_1} = 1$ . Moreover,  $i_1 \notin \text{IS}, i_2 \in \text{IS}$  and we are interested in bounding  $c(j, i_2)$  as a function of  $\alpha_j$ .

### 3.2.2 $k$ -Means objective in Euclidean metrics

We start with some intuition that illustrates our approach. From the standard analysis of JV (and our analysis of  $k$ -means in general metrics), it is clear that the bottleneck for the approximation guarantee comes from the connection-cost analysis of clients that need to do a “3-hop” as illustrated in the left part of Figure 1: client  $j$  is connected to open facility  $i_2$  and the squared-distance is bounded by the path  $j - i_1 - j_1 - i_2$ . Moreover, this analysis is tight when considering  $\text{JV} = \text{JV}(\infty)$ . Our strategy will now be as follows: Select  $\delta$  to be a constant smaller than 4. This means that in the configurations of Figure 1, we will also open  $i_2$  if the distance between  $i_1$  and  $i_2$  is close to 2. Therefore, if we do not open  $i_2$ , the distance between  $i_1$  and  $i_2$  is less than 2 (as in the right part of Figure 1) which allows us to get an approximation guarantee better than 9. However, this might result in a client contributing to the opening cost of many facilities in  $\text{IS}$ . Nonetheless, by using the properties of Euclidean metrics, we show that even in this case, we are able to achieve a LMP approximation guarantee with ratio better than 9.

Specifically, define  $\delta_{\text{mean}}$  to be the constant larger than 2 that minimizes

$$\rho_{\text{mean}}(\delta) = \max \left\{ (1 + \sqrt{\delta})^2, \frac{1}{\delta/2 - 1} \right\},$$

which will be our approximation guarantee. It can be verified that  $\delta_{\text{mean}} \approx 2.3146$  and  $\rho_{\text{mean}} \approx 6.3574$ . Let also  $c(j, i) = d(j, i)^2$  where  $d$  is the underlying Euclidean metric. The proof uses the following basic facts about squared-distances in Euclidean metrics: given  $x_1, x_2, \dots, x_s \in \mathbb{R}^\ell$ , we have that  $\min_{y \in \mathbb{R}^\ell} \sum_{i=1}^s \|x_i - y\|_2^2$  is attained by the *centroid*  $\mu = \frac{1}{s} \sum_{i=1}^s x_i$  and in addition we have the identity  $\sum_{i=1}^s \|x_i - \mu\|_2^2 = \frac{1}{2s} \sum_{i=1}^s \sum_{j=1}^s \|x_i - x_j\|_2^2$ .

**Theorem 3.4.** *Let  $d$  be a Euclidean metric on  $\mathcal{D} \cup \mathcal{F}$  and suppose that  $c(j, i) = d(j, i)^2$  for every  $i \in \mathcal{F}$  and  $j \in \mathcal{D}$ . Then, for any  $\lambda \geq 0$ , Algorithm  $\text{JV}(\delta_{\text{mean}})$  constructs a solution  $\alpha$  to  $\text{DUAL}(\lambda)$  and returns a set  $\text{IS}$  of opened facilities such that*

$$\sum_{j \in S} c(j, \text{IS}) \leq \rho_{\text{mean}} \cdot \left( \sum_{j \in \mathcal{D}} \alpha_j - \lambda |\text{IS}| \right).$$

*Proof.* To simplify notation, we use  $\delta$  instead of  $\delta_{\text{mean}}$  throughout the proof. Consider any client  $j \in \mathcal{D}$ . We shall prove that

$$\frac{c(j, \text{IS})}{\rho_{\text{mean}}} \leq \alpha_j - \sum_{i \in N(j) \cap \text{IS}} (\alpha_j - c(j, i)) = \alpha_j - \sum_{i \in \text{IS}} [\alpha_j - c(j, i)]^+.$$

Similarly to the proof of Theorem 3.3, the statement then follows by summing up over all clients. A difference compared to the standard analysis of JV is that in our algorithm we may open several facilities in  $N(j)$ , i.e., client  $j$  may contribute to the opening of several facilities. We divide our analysis into the three cases  $|N(j) \cap \text{IS}| = 1$ ,  $|N(j) \cap \text{IS}| > 1$ , and  $|N(j) \cap \text{IS}| = 0$ . For brevity, let  $S$  denote  $N(j) \cap \text{IS}$  and  $s = |S|$ .

Case  $s = 1$ : If we let  $i^*$  be the unique facility in  $S$ ,

$$\frac{c(j, \text{IS})}{\rho_{\text{mean}}} \leq c(j, \text{IS}) \leq c(j, i^*) = \alpha_j - (\alpha_j - c(j, i^*)) = \alpha_j - \sum_{i \in N(j) \cap \text{IS}} (\alpha_j - c(j, i)).$$

Case  $s > 1$ : In this case, there are multiple facilities in  $\text{IS}$  that  $j$  is contributing to. We need to show that  $\alpha_j - \sum_{i \in S} (\alpha_j - c(j, i)) \geq \frac{1}{\rho_{\text{mean}}} c(j, \text{IS})$ .

The sum  $\sum_{i \in S} c(j, i)$  is the sum of square distances from  $j$  to facilities in  $S$  which is at least the sum of square distances of these facilities from their centroid  $\mu$ , i.e.,  $\sum_{i \in S} c(j, i) \geq \sum_{i \in S} c(i, \mu)$ . Moreover, by the identity,  $\sum_{i \in S} c(i, \mu) = \frac{1}{2s} \sum_{i, i' \in S} c(i, i')$ , we get

$$\sum_{i \in S} c(j, i) \geq \frac{1}{2s} \sum_{i, i' \in S} c(i, i').$$

As there is no edge between any pair of facilities in  $S \subseteq \text{IS}$ , we must have

$$c(i, i') > \delta \cdot \min(t_i, t_{i'}) \geq \delta \cdot \alpha_j,$$

where the last inequality follows because  $j$  is contributing to both  $i$  and  $i'$  and hence  $\min(t_i, t_{i'}) \geq \alpha_j$ . By the above,

$$\sum_{i \in S} c(j, i) \geq \frac{\sum_{i, i' \in S} c(i, i')}{2s} \geq \frac{\sum_{i \neq i' \in S} \delta \cdot \alpha_j}{2s} = \delta \cdot \frac{s-1}{2} \cdot \alpha_j.$$

Hence,

$$\sum_{i \in S} (\alpha_j - c(j, i)) \leq \left( s - \delta \cdot \frac{s-1}{2} \right) \alpha_j = \left( s(1 - \frac{\delta}{2}) + \frac{\delta}{2} \right) \alpha_j.$$

Now, since  $\delta \geq 2$  the above upper bound is a non-increasing function of  $s$ . Therefore, since  $s \geq 2$  we always have

$$\sum_{i \in S} (\alpha_j - c(j, i)) \leq (2 - \frac{\delta}{2}) \alpha_j. \quad (3.3)$$

We also know that  $\alpha_j > c(j, i)$  for any  $i \in S$ . Therefore,  $\alpha_j > c(j, \text{IS})$  and, since  $\delta \geq 2$ :

$$(\frac{\delta}{2} - 1) c(j, \text{IS}) \leq (\frac{\delta}{2} - 1) \alpha_j. \quad (3.4)$$

Combining Inequalities (3.3) and (3.4),

$$\sum_{i \in S} (\alpha_j - c(j, i)) + (\frac{\delta}{2} - 1) c(j, \text{IS}) \leq (2 - \frac{\delta}{2}) \alpha_j + (\frac{\delta}{2} - 1) \alpha_j = \alpha_j.$$

We conclude the analysis of this case by rearranging the above inequality and recalling that  $\rho_{\text{mean}} \geq \frac{1}{\delta/2-1}$ .

Case  $s = 0$ : Here, we claim that there exists a tight facility  $i$  such that

$$(1 + \sqrt{\delta})\sqrt{\alpha_j} \geq d(j, i) + \sqrt{\delta t_i}. \quad (3.5)$$

To see that such a facility  $i$  exists, consider the witness  $w(j)$  of  $j$ . By Claim 3.1, we have  $\alpha_j \geq t_{w(j)}$  and since  $j$  has a tight edge to its witness  $w(j)$ ,  $\alpha_j \geq c(j, w(j)) = d(j, w(j))^2$ ; or, equivalently,  $\sqrt{\alpha_j} \geq \sqrt{t_{w(j)}}$  and  $\sqrt{\alpha_j} \geq d(j, w(j))$  which implies that there is a tight facility, namely  $w(j)$ , satisfying (3.5).

Since  $\mathbf{IS}$  is a maximal independent set of  $H$ , either  $i \in \mathbf{IS}$ , in which case  $d(j, \mathbf{IS}) \leq d(j, i)$ , or there is an  $i' \in \mathbf{IS}$  such that the edge  $(i', i)$  is in  $H$ , in which case

$$d(j, \mathbf{IS}) \leq d(j, i) + d(i, i') \leq d(j, i) + \sqrt{\delta t_i},$$

where the inequality follows from  $d(i, i')^2 = c(i, i') \leq \delta \min(t_i, t_{i'})$  by the definition of  $H$ . In any case, we have by (3.5)

$$d(j, \mathbf{IS}) \leq (1 + \sqrt{\delta})\sqrt{\alpha_j}.$$

Squaring both sides and recalling that  $\rho_{\text{mean}} \geq \frac{1}{(1+\sqrt{\delta})^2}$  completes the last case and the proof of the theorem.  $\square$

### 3.2.3 $k$ -Median objective in Euclidean metrics

We use a very similar approach as the one for  $k$ -means (in Euclidean metrics) to address the  $k$ -median objective in Euclidean metrics. In this section, we have  $c(j, i) = d(j, i)$ , i.e., the distances are *not* squared. Define,

$$\delta_{\text{med}} = \sqrt{\frac{8}{3}} \quad \text{and} \quad \rho_{\text{med}} = 1 + \sqrt{\frac{8}{3}} = \max \left( 1 + \delta_{\text{med}}, 1/(\delta_{\text{med}} - 1), 1/(\frac{3}{2}\delta_{\text{med}} - 2) \right).$$

We have  $\delta_{\text{med}} \approx 1.633$  and  $\rho_{\text{med}} \approx 2.633$ .

**Theorem 3.5.** *Let  $d$  be a Euclidean metric on  $\mathcal{D} \cup \mathcal{F}$  and suppose that  $c(j, i) = d(j, i)$  for every  $i \in \mathcal{F}$  and  $j \in \mathcal{D}$ . Then, for any  $\lambda \geq 0$ , Algorithm  $\mathcal{JV}(\delta_{\text{med}})$  constructs a solution  $\alpha$  to  $\text{DUAL}(\lambda)$  and returns a set  $\mathbf{IS}$  of opened facilities such that*

$$\sum_{j \in S} c(j, \mathbf{IS}) \leq \rho_{\text{med}} \cdot \left( \sum_{j \in \mathcal{D}} \alpha_j - \lambda |\mathbf{IS}| \right).$$

*Proof.* To simplify notation, we use  $\delta$  instead of  $\delta_{\text{med}}$  throughout the proof. Similar to the proof of the previous theorem, we proceed by considering a single client  $j$  and prove

$$\frac{c(j, \mathbf{IS})}{\rho_{\text{med}}} \leq \alpha_j - \sum_{i \in N(j) \cap \mathbf{IS}} (\alpha_j - c(j, i)) = \alpha_j - \sum_{i \in \mathbf{IS}} [\alpha_j - c(j, i)]^+.$$

Let  $S$  denote  $N(j) \cap \mathbf{IS}$  and  $s = |S|$ . We again proceed by case distinction on  $s$ . We first bound the number of cases.

**Claim 3.6.** *We have  $s \leq 3$ .*

*Proof.* Using the centroid property of squared distances in Euclidean space,

$$\sum_{j \in S} d(j, i)^2 \geq \frac{\sum_{i, i' \in S} d(i, i')^2}{2s} > \frac{\delta^2(s-1)\alpha_j^2}{2},$$

where the last inequality follows from that  $i, i' \in \text{IS}$  are not adjacent in  $H$  so  $d(i, i') > \delta \min(t_i, t_{i'})$  and  $\min(t_i, t_{i'}) \geq \alpha_j$  since  $i, i' \in S$ . Since the left-hand-side is upper bounded by  $s\alpha_j^2$ , we get  $s > \frac{\delta^2(s-1)}{2}$ . Therefore  $s < \frac{\delta^2}{\delta^2-2} = 4$ .  $\square$

We now proceed by considering the cases  $s = 0, 1, 2, 3$ .

Case  $s = 0$ : Consider the witness  $i_1$  of  $j$ . We have  $\alpha_j \geq t_{i_1}$  and  $\alpha_j \geq c(j, i_1)$ . Since  $\text{IS}$  is a maximal independent set of  $H$ , either  $i_1 \in \text{IS}$ , in which case  $c(j, \text{IS}) = d(j, \text{IS}) \leq d(j, i_1) \leq \alpha_j$ , or there is an  $i_2 \in \text{IS}$  such that the edge  $(i_1, i_2)$  is in  $H$ , in which case

$$c(j, \text{IS}) = d(j, \text{IS}) \leq d(j, i_1) + d(i_1, i_2) \leq d(j, i_1) + \delta t_{i_1} \leq (1 + \delta)\alpha_j.$$

In any case, we have  $c(j, \text{IS})/\rho_{\text{med}} \leq \alpha_j$  as required.

Case  $s = 1$ : If we let  $i^*$  be the unique facility in  $S$ ,

$$\frac{c(j, \text{IS})}{\rho_{\text{med}}} \leq c(j, \text{IS}) \leq c(j, i^*) = \alpha_j - (\alpha_j - c(j, i^*)) = \alpha_j - \sum_{i \in N(j) \cap \text{IS}} (\alpha_j - c(j, i)).$$

Case  $s = 2$ : Let  $S = \{i_1^*, i_2^*\}$ . We have

$$\begin{aligned} 2\alpha_j &= c(j, i_1^*) + c(j, i_2^*) + (\alpha_j - c(j, i_1^*)) + (\alpha_j - c(j, i_2^*)) \\ &\geq c(i_1^*, i_2^*) + \sum_{i \in S} (\alpha_j - c(j, i)) \\ &\geq \delta\alpha_j + \sum_{i \in S} (\alpha_j - c(j, i)), \end{aligned}$$

where we used the triangle inequality and that  $c(i_1^*, i_2^*) > \delta \min(t_{i_1^*}, t_{i_2^*}) \geq \delta\alpha_j$  since both  $i_1^*$  and  $i_2^*$  are in  $S$  and hence  $i_1^*$  and  $i_2^*$  are not adjacent in  $H$ . Rearranging the above inequality, we have

$$(\delta - 1)c(j, \text{IS}) \leq \alpha_j - \sum_{i \in S} (\alpha_j - c(j, i)),$$

and the case follows because  $\rho_{\text{med}} \geq 1/(\delta - 1)$ .

Case  $s = 3$ : Similar to the previous case,

$$\begin{aligned} 3\alpha_j &= \sum_{i \in S} c(j, i) + \sum_{i \in S} (\alpha_j - c(j, i)) \\ &\geq \frac{1}{2} \sum_{\{i, i'\} \subseteq S} c(i, i') + \sum_{i \in S} (\alpha_j - c(j, i)) \\ &\geq \frac{3\delta}{2} \cdot \alpha_j + \sum_{i \in S} (\alpha_j - c(j, i)), \end{aligned}$$

using the triangle inequality. Rearranging the above inequality, we have

$$\left(\frac{3\delta}{2} - 2\right) c(j, \text{IS}) \leq \alpha_j - \sum_{i \in S} (\alpha_j - c(j, i))$$

and the lemma follows because  $\rho_{\text{med}} \geq 1/(\frac{3\delta}{2} - 2)$ .  $\square$

## 4 Quasi-polynomial time algorithm

In this section, we present a quasi-polynomial time algorithm that achieves the improved approximation guarantees explained in the previous section. We also introduce several of the ideas used in the polynomial time algorithm. Although the results obtained in this section are weaker (quasi-polynomial instead of polynomial), we believe that the easier quasi-polynomial algorithm serves as a good starting point before reading the more complex polynomial time algorithm. From now on, we concentrate on the  $k$ -means problem and we let  $\rho = \rho_{\text{mean}}$  denote the approximation guarantee and  $\delta = \delta_{\text{mean}}$  denote the parameter to our algorithm, where  $\rho_{\text{mean}}$  and  $\delta_{\text{mean}}$  are defined as in Section 3.2.2 (it will be clear that the techniques presented here are easily applicable to the other considered objectives, as well). Throughout this section we fix  $\epsilon > 0$  to be a small constant, and we assume for notational convenience and without loss of generality that  $n \gg 1/\epsilon$ . We shall also assume that the distances satisfy the following:

**Lemma 4.1.** *By losing a factor  $(1 + 100/n^2)$  in the approximation guarantee, we can assume that the squared-distance between any client  $j$  and any facility  $i$  satisfies:  $1 \leq d(i, j)^2 \leq n^6$ , where  $n = |\mathcal{D}|$ .*

The proof follows by standard discretization techniques and is presented in Appendix B.

Our algorithm will produce a  $(\rho + O(\epsilon))$ -approximate solution. In the algorithm, we consider separately the two phases of the primal-dual algorithm from Section 3.2.2. Suppose that the first phase produces a set of values  $\alpha = \{\alpha_j\}_{j \in \mathcal{D}}$  satisfying the following definition:

**Definition 4.2.** *A feasible solution  $\alpha$  of  $\text{DUAL}(\lambda)$  is good if for every  $j \in \mathcal{D}$  there exists a tight facility  $i$  such that  $(1 + \sqrt{\delta} + \epsilon)\sqrt{\alpha_j} \geq d(j, i) + \sqrt{\delta}t_i$ .*

Recall that for a dual solution  $\alpha$ ,  $t_i$  is defined to be the largest  $\alpha$ -value out of all clients that are contributing to a facility  $i$ :  $t_i = \max_{j \in N(i)} \alpha_j$  where  $N(i) = \{j \in \mathcal{D} : \alpha_j - d(i, j)^2 > 0\}$ .

As the condition of Definition 4.2 relaxes (3.5) by a tiny amount (regulated by  $\epsilon$ ), our analysis in Section 3 shows that as long as the first stage of the primal-dual algorithm produces an  $\alpha$  that is good, the second stage will find a set of facilities  $\text{IS}$  such that  $\sum_{j \in \mathcal{D}} d(j, \text{IS})^2 = \sum_{j \in \mathcal{D}} c(j, \text{IS}) \leq (\rho + O(\epsilon))(\sum_{j \in \mathcal{D}} \alpha_j - \lambda|\text{IS}|)$ . If we could somehow find a value  $\lambda$  such that the second stage opened *exactly*  $k$  facilities, then we would obtain a  $(\rho + O(\epsilon))$ -approximation algorithm. In order to accomplish this, we first enumerate all potential values  $\lambda = 0, 1 \cdot \epsilon_z, 2 \cdot \epsilon_z, \dots, L \cdot \epsilon_z$ , where  $\epsilon_z$  is a small step size and  $L$  is large enough to guarantee that we eventually find a solution of size at most  $k$  (for a precise definition of  $L$  and  $\epsilon_z$ , see (4.1) and (4.2)). Specifically, in Section 4.1, we give an algorithm that in time  $n^{O(\epsilon^{-1} \log n)}$  generates a quasi-polynomial-length sequence of solutions  $\alpha^{(0)}, \alpha^{(1)}, \dots, \alpha^{(L)}$ , where  $\alpha^{(\ell)}$  is a good solution to  $\text{DUAL}(\ell \cdot \epsilon_z)$ . We shall ensure that each consecutive set of values  $\alpha^{(\ell)}, \alpha^{(\ell+1)}$  are *close* in the following sense:

**Definition 4.3.** *Two solutions  $\alpha$  and  $\alpha'$  are close if  $|\alpha'_j - \alpha_j| \leq \frac{1}{n^2}$  for all  $j \in \mathcal{D}$ .*

Unfortunately, it may be the case that for a good solution  $\alpha^{(\ell)}$  to  $\text{DUAL}(\lambda)$ , the second stage of our algorithm opens more than  $k$  facilities, while for a good solution  $\alpha^{(\ell+1)}$  to  $\text{DUAL}(\lambda + \epsilon_z)$ , it opens fewer than  $k$  facilities. In order to obtain a solution that opens *exactly*  $k$  facilities, we must somehow interpolate between consecutive solutions in our sequence. In Section 4.2 we describe an algorithm that accomplishes this task. Specifically, for each pair of consecutive solutions  $\alpha^{(\ell)}, \alpha^{(\ell+1)}$  we show that, since their  $\alpha$ -values are nearly the same, we can control the way in which a maximal independent set in the associated conflict graphs changes. Formally, we show how to maintain a sequence of approximate integral solutions with cost bounded by  $\alpha^{(\ell)}$  and  $\alpha^{(\ell+1)}$ , in which the number of open facilities decreases by at most one in each step. This ensures that some solution indeed opens exactly  $k$  facilities and it will be found in time  $n^{O(\epsilon^{-1} \log n)}$ .

## 4.1 Generating a sequence of close, good solutions

We first describe our procedure for generating a close sequence of good solutions. Select the following parameters

$$\epsilon_z = n^{-3-10\log_{1+\epsilon} n}, \quad (4.1)$$

$$L = 4n^7 \cdot \epsilon_z^{-1} = n^{O(\epsilon^{-1} \log n)}. \quad (4.2)$$

We now describe a procedure **QUASISWEEP** that takes as input a good dual solution  $\alpha^{\text{in}}$  of  $\text{DUAL}(\lambda)$  and outputs a good dual solution  $\alpha^{\text{out}}$  of  $\text{DUAL}(\lambda + \epsilon_z)$  such that  $\alpha^{\text{in}}$  and  $\alpha^{\text{out}}$  are close. In order to generate the desired close sequence of solutions, we first define an initial solution for  $\text{DUAL}(0)$  by  $\alpha_j = \min_{i \in \mathcal{F}} d(i, j)^2$  for  $j \in \mathcal{D}$ . Then, for  $0 \leq \ell < L$ , we call **QUASISWEEP** with  $\alpha^{\text{in}} = \alpha^{(\ell)}$  to generate the next solution  $\alpha^{(\ell+1)}$  in our sequence. We shall show that each  $\alpha^{(\ell)}$  is a feasible dual solution of  $\text{DUAL}(\ell \cdot \epsilon_z)$ , and that the following invariant holds throughout the generation of our sequence:

**Invariant 1.** *In every solution  $\alpha^{(\ell)}$ , ( $0 \leq \ell \leq L$ ), every client  $j \in \mathcal{D}$  has a tight edge to a tight facility  $w(j) \in \mathcal{F}$  (its witness) such that  $t_{w(j)} \leq (1 + \epsilon)\alpha_j$ .*

Note that this implies that each solution in our sequence is good. Indeed, consider a dual solution  $\alpha$  that satisfies Invariant 1. Then, for any client  $j$ , we have some  $i (= w(j))$  such that  $\sqrt{\alpha_j} \geq d(i, j)$  (since  $j$  has a tight edge to  $w(j)$ ) and  $\sqrt{(1 + \epsilon)\delta\alpha_j} \geq \sqrt{\delta t_i}$ . Hence,

$$(1 + \sqrt{\delta} + \epsilon)\sqrt{\alpha_j} \geq \left(1 + \sqrt{(1 + \epsilon)\delta}\right) \sqrt{\alpha_j} \geq d(i, j) + \sqrt{\delta t_i},$$

and so  $\alpha$  is good (here, for the first inequality we have used that  $\sqrt{1 + \epsilon} \leq 1 + \epsilon/2$  and  $\sqrt{\delta} \leq 2$ ). We observe that our initial solution  $\alpha^{(0)}$  has  $t_i = 0$  for all  $i \in \mathcal{F}$ , and so Invariant 1 holds trivially. In our following analysis, we will show that each call to **SWEEP** preserves Invariant 1.

We will use the notion of *buckets* that partition the real line:

**Definition 4.4.** *For any value  $v \in \mathbb{R}$ , let*

$$B(v) = \begin{cases} 0 & \text{if } v < 1, \\ 1 + \lfloor \log_{1+\epsilon}(v) \rfloor & \text{if } v \geq 1. \end{cases}$$

*We say that  $B(v)$  is the index of the bucket containing  $v$ .*

The buckets will be used to partition the  $\alpha$ -values of the clients. Since  $\alpha_j$  will always be at least 1 for each client (because its distance to any facility is at least 1), the definition gives the property that the  $\alpha$ -values of any two clients  $j$  and  $j'$  in the same bucket differ by at most a factor of  $1 + \epsilon$ .

### 4.1.1 Description of **QUASISWEEP**

We now formally describe the procedure **QUASISWEEP** that, given the last previously generated solution  $\alpha^{\text{in}}$  in our sequences produces a solution  $\alpha^{\text{out}}$  returned next.

We initialize the algorithm by setting  $\alpha_j = \alpha_j^{\text{in}}$  for each  $j \in \mathcal{D}$  and by increasing the opening prices of each facility from  $\lambda$  to  $\lambda + \epsilon_z$ . At this point, no facility is tight and therefore the solution  $\alpha$  is not a good solution of  $\text{DUAL}(\lambda + \epsilon_z)$ . We now describe how to modify  $\alpha$  to obtain a solution  $\alpha^{\text{out}}$  satisfying Invariant 1 (and hence into a good solution). The algorithm will maintain a current set  $A$  of active clients and a current threshold  $\theta$ . Initially,  $A = \emptyset$ , and  $\theta = 0$ . We slowly increase  $\theta$  and whenever  $\theta = \alpha_j$  for some client  $j$ , we add  $j$  to  $A$ . While  $j \in A$ , we increase  $\alpha_j$  at the same rate as  $\theta$ . We remove a client  $j$  from  $A$ , whenever the following occurs:

$j$  has a tight edge to some tight facility  $i$  with  $B(\alpha_j) \geq B(t_i)$ . In this case, we say that  $i$  is the *witness* of  $j$ .

Note that if a client  $j$  satisfies this condition when it is added to  $A$ , then we remove  $j$  from  $A$  immediately after it is added. Then,  $\alpha_j$  is not increased.

Increasing the  $\alpha$ -values for clients in  $A$ , may cause the contributions to some facility  $i$  to exceed the opening cost  $\lambda + \epsilon_z$ . To prevent this from happening, we also decrease every value  $\alpha_j$  with  $B(\alpha_j) > B(\theta)$  at a rate of  $|A|$  times the rate that  $\theta$  is increasing. Observe that while there exists any such  $j \in N(i)$ , the total contribution of the clients toward opening this  $i$  cannot increase, and so  $i$  cannot become tight. It follows that once any facility  $i$  becomes tight,  $B(\alpha_j) \leq B(\theta)$  for every  $j \in N(i)$  and so  $i$  is presently a witness for all clients  $j \in N(i) \cap A$ . As no such client will subsequently decrease,  $i$  remains tight until the end of QUASISWEEP. Moreover, any client  $j'$  that is added to  $A$  later will immediately be removed from  $A$  as soon as it has a tight edge to  $i$ . Thus, neither the total contribution to  $i$  nor  $t_i$  can subsequently increase and so  $i$  remains a witness for all such  $j$  for the remainder of QUASISWEEP.

We stop increasing  $\theta$  once every client  $j$  has been added and removed from  $A$ . The procedure QUASISWEEP then terminates and outputs  $\alpha^{\text{out}} = \alpha$ . As we have just argued, the contributions to any tight facility can never increase, and every client that is removed from  $j$  will have a witness through the rest of QUASISWEEP (in particular, in  $\alpha^{\text{out}}$ ). Thus,  $\alpha^{\text{out}}$  is a feasible solution of  $\text{DUAL}(\lambda + \epsilon_z)$  in which every client has a witness, and as the values in the same bucket differ by at most a factor  $1 + \epsilon$  (using that the  $\alpha$ -value of each client is at least 1), the output of SWEEP always satisfies Invariant 1.

This completes the description of QUASISWEEP. For a small example of its execution see Figure 2. We now proceed to its analysis.

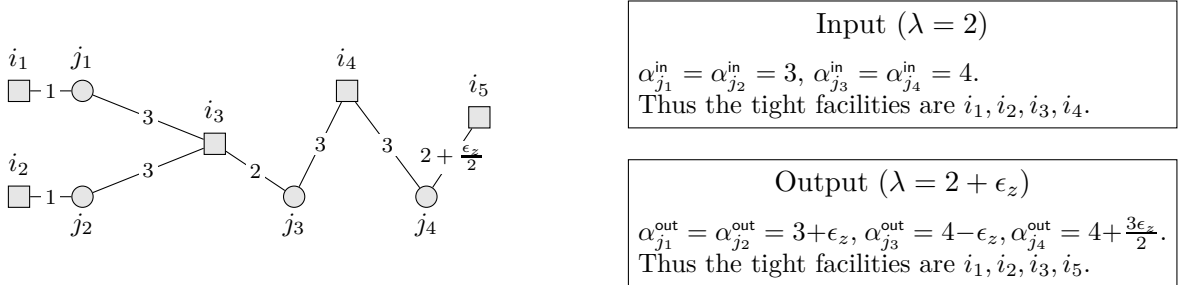


Figure 2: The instance has 4 clients and 5 facilities depicted by circles and squares, respectively. The number on an edge is the squared-distance of that edge and the squared-distances that are not depicted are all of value 5. Given the input solution  $\alpha^{\text{in}}$  with  $\lambda = 2$ , QUASISWEEP proceeds as follows. First the opening prices of facilities are increased to  $2 + \epsilon_z$ . Next the clients  $j_1, j_2$  are added to the set  $A$  of active clients when the threshold  $\theta = 3$ . Then, until  $\theta = 3 + \epsilon_z$ ,  $\alpha_{j_1}$  and  $\alpha_{j_2}$  increase at a uniform rate while the (significantly) larger dual values  $\alpha_{j_3}$  and  $\alpha_{j_4}$  are decreasing  $|A| = 2$  times that rate. At the point  $\theta = 3 + \epsilon_z$ , both  $i_1$  and  $i_2$  become tight and the witnesses of  $j_1$  and  $j_2$  respectively. This causes these clients to be removed from  $A$  which stops their increase and the decrease of the larger values. When  $\theta = 4 - 2\epsilon_z$ ,  $j_3$  and  $j_4$  are added to  $A$  and they start to increase at a uniform rate. Next, the facility  $i_3$  becomes tight when  $\theta = 4 - \epsilon_z$  and client  $j_3$  is removed from  $A$  with  $i_3$  as its witness. Finally,  $j_4$  is removed from  $A$  when  $\theta = 4 + 3\epsilon_z/2$  at which point  $i_5$  becomes tight and its witness.

### 4.1.2 Closeness and running time

We begin by showing that QUASISWEEP produces a close sequence of solutions.

**Lemma 4.5.** *For each client  $j \in \mathcal{D}$ , we have  $|\alpha_j^{\text{in}} - \alpha_j^{\text{out}}| \leq 1/n^2$ .*

*Proof.* We first note that the largest  $\alpha$ -value at any time is at most  $(\lambda + \epsilon_z) + n^6 \leq L\epsilon_z + n^6 = 4n^7 + n^6 \leq 5n^7$ . This follows from the feasibility of  $\alpha$  because, by Lemma 4.1, no squared-distance is larger than  $n^6$  and the opening cost of any facility is at most  $\lambda + \epsilon_z \leq L\epsilon_z$ . Hence,  $B(\alpha_j) \leq 1 + \lfloor \log_{1+\epsilon}(5n^7) \rfloor \leq 10 \log_{1+\epsilon}(n)$  for any client  $j$  and dual solution  $\alpha$ .

We now prove the following claim:

**Claim.** *Any  $\alpha_j$  can increase by at most  $\epsilon_z n^{3b}$  while  $B(\theta) \leq b$ .*

*Proof.* The proof is by induction on  $b = 0, 1, \dots, 10 \log_{1+\epsilon}(n)$ .

**Base case  $b = 0$ :** This case is trivially true because there are no clients  $j$  with  $B(\alpha_j) = 0$ , and so no clients can be added to  $A$  while  $B(\theta) = 0$ . Indeed, any client  $j$  had a tight edge to some facility in  $\alpha^{\text{in}}$ , which by Lemma 4.1 implies  $\alpha_j^{\text{in}} \geq 1$ , and a client's  $\alpha$ -value can decrease only while some smaller  $\alpha$ -value is increasing.

**Inductive step (assume true for  $0, 1, \dots, b-1$  and prove for  $b$ ):** Now, we suppose some  $\alpha_j$  is increasing while  $B(\theta) \leq b$ . Note that we then must have  $\alpha_j = \theta$ . Let  $i$  be the witness of  $j$  in  $\alpha^{\text{in}}$ , and let  $N^{\text{in}}(i)$  be the set of clients contributing to  $i$  at this time. We further suppose that  $\alpha_j$  is increased by at least  $\epsilon_z$  while  $B(\theta) \leq b$ ; otherwise the claim follows immediately, since  $\epsilon_z \leq n^{3b}\epsilon_z$  for all  $b \geq 0$ .

First, suppose that  $\alpha_j < \alpha_j^{\text{in}}$ . Then,  $\alpha_j$  was previously decreased. Moreover, since  $\alpha_j$  was increased by at least  $\epsilon_z$  while  $B(\theta) \leq b$ , we must have previously decreased  $\alpha_j$  while  $B(\alpha_j) \leq b$ . In particular, at the last moment  $\alpha_j$  was decreased, we must have had  $B(\alpha_j) \leq b$ , and since  $\alpha_j$  was decreasing at this moment, we also had  $B(\theta) < B(\alpha_j)$ . Therefore  $\alpha_j$  was decreased only while  $B(\theta) < b$ . Moreover, during this time,  $j$ 's  $\alpha$ -value was decreased at a rate of  $|A|$ , and so was decreased at most  $n$  times the amount that any other client was increased. By the induction hypothesis, this increase was at most  $\epsilon_z \cdot n^{3b-3}$ , and so  $\alpha_j$  can be increased at most  $\epsilon_z \cdot n^{3b-2}$  until  $\alpha_j = \alpha_j^{\text{in}}$ .

Now, we consider how much  $\alpha_j$  may increase while  $\alpha_j \geq \alpha_j^{\text{in}}$ . For each  $j' \in N^{\text{in}}(i)$  we must have initially had  $B(\alpha_{j'}^{\text{in}}) \leq B(\alpha_j^{\text{in}}) \leq b$  since  $i$  is a witness for  $j$  in  $\alpha^{\text{in}}$ , and so the  $\alpha$ -value of  $j'$  was decreased by QUASISWEEP only while  $B(\theta) \leq b-1$ . Then, by the same argument as above, the  $\alpha$ -value of any  $j' \in N^{\text{in}}(i)$  can decrease at most  $\epsilon_z n^{3b-2}$  throughout QUASISWEEP. Thus, the total contribution to  $i$  from all  $j' \neq j$  can decrease at most  $(n-1) \cdot \epsilon_z \cdot n^{3b-2}$ . After increasing  $\alpha_j$  at most  $(n-1) \cdot \epsilon_z \cdot n^{3b-2} + \epsilon_z$ ,  $i$  will again be tight. Moreover, at this moment, any client  $j'$  contributing to  $i$  was either already added to  $A$  (and potentially also removed) in which case  $\alpha_{j'} \leq \theta = \alpha_j$  or it was not already added to  $A$  in which case  $\alpha_{j'} \leq \alpha_j^{\text{in}}$ . Hence, as  $B(\alpha_{j'}^{\text{in}}) \leq B(\alpha_j^{\text{in}}) \leq B(\alpha_j)$ ,  $i$  is a witness for  $j$ , and  $j$  will be removed from  $A$ .

Altogether, the total amount  $\alpha_j$  can increase while  $B(\theta) \leq b$  is then the sum of these two increases, which is  $\epsilon_z \cdot n^{3b-2} + (n-1) \cdot \epsilon_z \cdot n^{3b-2} + \epsilon_z \leq \epsilon_z \cdot n^{3b}$ , as required.  $\square$

The claim immediately bounds the increase  $\alpha_j^{\text{out}} - \alpha_j^{\text{in}}$  by  $\frac{1}{n^3} \leq \frac{1}{n^2}$  as required (recall that  $\epsilon_z = n^{-3-10 \log_{1+\epsilon} n}$ ). Moreover, as shown in the proof of the claim above, the  $\alpha$ -value of every client decreases by no more than  $n$  times the maximum increase in the  $\alpha$ -value of any client. Then, the desired bound  $\frac{1}{n^2}$  on  $\alpha_j^{\text{in}} - \alpha_j^{\text{out}}$  follows as well.  $\square$



**Running time analysis** The procedure QUASISWEEP was presented in a continuous fashion. We now describe a polynomial time implementation of QUASISWEEP. As presented, QUASISWEEP maintains only the  $\alpha$ -values of each client, the value of  $\theta$ , and the set  $A$  of active clients. We suppose we are increasing  $\theta$  at the speed of 1, so that the value of  $\theta$  corresponds to the current time. Then, QUASISWEEP changes each  $\alpha$ -value at the speed of either 0, 1, or  $-|A|$ . Moreover, this speed does not change until one of the following events happens:

- Event 1: Client  $j$  joins  $A$ : this can happen only if  $\alpha_j = \theta$ .
- Event 2:  $\theta$  changes buckets: this can only happen when  $\theta$  has reached the border of a bucket.
- Event 3: Facility  $i$  becomes tight: this can happen if (1) no client with a tight edge to  $i$  is decreasing, and (2) some client in  $A$  has a tight edge to  $i$ .
- Event 4: Client  $j$  gains a tight edge to facility  $i$ : this can happen only if  $j \in A$ .
- Event 5: Client  $j \notin A$  changes bucket and enters the same bucket as  $\theta$ : this can happen only if  $\alpha_j$  is being decreased.

Note that we remove a client from  $A$  either immediately after it is added to  $A$ , at the time that some facility becomes tight, or at the time that it gains a tight edge to some (tight) facility. Therefore, we do not need to add an event for removing a client from  $A$ , since it only happens if one of the above events happen.

The polynomial time QUASISWEEP now works as follows: In each step, we find the next time that any one of the above events happens, then increase/decrease each  $\alpha$ -value according to its current speed to obtain a new set of values at this time. Then, we update  $\theta$ ,  $A$ , and our set of speeds and continue. We need to show how we can efficiently compute the next event that happens, and also we need to prove that the number of such events is polynomial. In what follows, we compute the time until each event above happens, assuming that it is the next event that happens. Then the next event that actually happens is the event with the minimum such time (breaking ties arbitrarily).

We now consider each of the above events in turn:

- The time until the Event 1 may happen next is  $\min_{\alpha_j > \theta} \alpha_j - \theta$ . Also we have exactly  $n$  occurrences of this event.
- The time until Event 2 may happen next is the difference between  $\theta$  and the border of the next bucket, i.e.,  $B_{next} - \theta$ , where  $B_{next} = (1 + \epsilon)^{B(\theta)}$ . We have at most  $O(\epsilon^{-1} \log(n))$  such events.
- For Event 3, if some client with a tight edge to  $i$  is decreasing then (non-tight) facility  $i$  cannot become tight (due to the choice of the speed of decrease). If no decreasing client has a tight edge to facility  $i$ , then the time that  $i$  may become tight is

$$\frac{(\lambda + \epsilon_z) - \sum_{j \in \mathcal{D}} [\alpha_j - d(i, j)^2]^+}{|N(i) \cap A|}.$$

Notice that the numerator is the current slack of facility  $i$  and the denominator is the speed at which this slack decreases. Moreover, there are at most  $m = |\mathcal{F}|$  such events, since if a facility becomes tight, it will stay tight (as we discussed in our description of QUASISWEEP).

- The time until Event 4 may happen for some edge  $(j, i)$  is  $d(j, i)^2 - \alpha_j$  if  $\alpha_j < d(j, i)^2$ , and there are at most  $nm$  such events, since if an edge becomes tight, it remains tight afterwards.

- Finally, Event 5 may happen only for those clients  $j$  with  $B(\alpha_j) > B(\theta)$ . For any such  $j$ , the time until Event 5 happens is  $(\alpha_j - B_{next})/|A|$ . This event can happen also at most  $n$  times, since once  $B(\alpha_j) = B(\theta)$ ,  $j$  is no longer decreased. Note that when  $\alpha_j$  is decreasing, we consider it to change buckets at the moment that it lies on the lower border of its current bucket (i.e., at the moment that  $1 + \log_{1+\epsilon} \alpha_j = B_{next}$ ). It is easy to verify that still  $(1 + \epsilon)\alpha_j \leq \alpha_{j'}$  for any  $j$  and  $j'$  placed in the same bucket by this rule.

From the above, it is clear that the number of events are polynomial, and also that the next event can be computed in polynomial time.

We conclude the analysis of this section by noting that, as SWEEP is repeated  $L = n^{O(\epsilon^{-1} \log n)}$  times, the total running time for producing the sequence  $\alpha^{(0)}, \alpha^{(1)}, \dots, \alpha^{(L)}$  is  $n^{O(\epsilon^{-1} \log n)}$ .

## 4.2 Finding a solution of size $k$

In this section we describe our algorithm for finding a solution of  $k$  facilities given a close sequence  $\alpha^{(0)}, \alpha^{(1)}, \dots, \alpha^{(L)}$ , where  $\alpha^{(\ell)}$  is a good solution to  $\text{DUAL}(\epsilon_z \cdot \ell)$ .

We associate with each dual solution  $\alpha^{(\ell)}$  a client-facility graph and a conflict graph that are defined in exactly the same way as in Section 3.1: that is, the graph  $G^{(\ell)}$  is a bipartite graph with all of  $\mathcal{D}$  on one side and every tight facility in  $\alpha^{(\ell)}$  on the other and  $G^{(\ell)}$  contains the edge  $(j, i)$  if and only if  $\alpha_j^{(\ell)} > c(j, i)$ . Given each  $G^{(\ell)}$ , recall that  $H^{(\ell)}$  is then a graph consisting of the facilities present in  $G^{(\ell)}$ , which contains an edge  $(i, i')$  if and only if  $c(i, i') < \delta \min(t_i^{(\ell)}, t_{i'}^{(\ell)})$ , where for each  $i$ , we have  $t_i^{(\ell)} = \max\{\alpha_j^{(\ell)} : \alpha_j^{(\ell)} > c(j, i)\}$  (and again we adopt the convention that  $t_i^{(\ell)} = 0$  if  $\alpha_j^{(\ell)} \leq c(j, i)$  for all  $j \in \mathcal{D}$ ). Thus, we have a sequence  $G^{(0)}, \dots, G^{(L)}$  of client-facility graphs and a sequence  $H^{(0)}, \dots, H^{(L)}$  of conflict graphs obtained from our sequence of dual solutions. The main goal of this section is to give a corresponding sequence of maximal independent sets of the conflict graphs so that the size of the solution (independent set) never decreases by more than 1 in this sequence. Unfortunately, this is not quite possible. Instead, starting with a maximal independent set  $\text{IS}^{(\ell)}$  of  $H^{(\ell)}$ , we shall slowly transform it into a maximal independent set  $\text{IS}^{(\ell+1)}$  of  $H^{(\ell+1)}$  by considering maximal independent sets in a sequence of polynomially many intermediate conflict graphs  $H^{(\ell)} = H^{(\ell,0)}, H^{(\ell,1)}, \dots, H^{(\ell,p_\ell)} = H^{(\ell+1)}$ . We shall refer to these independent sets as  $\text{IS}^{(\ell)} = \text{IS}^{(\ell,0)}, \text{IS}^{(\ell,1)}, \dots, \text{IS}^{(\ell,p_\ell)} = \text{IS}^{(\ell+1)}$ . This interpolation will allow us to ensure that the size of our independent set decreases by at most 1 throughout this sequence. It follows that at some point we find a solution  $\text{IS}$  of size exactly  $k$ : On the one hand, since  $H^{(0)}$  contains all facilities and no edges we have  $\text{IS}^{(0)} = \mathcal{F}$ , which by assumption is strictly greater than  $k$ . On the other hand, we must have  $|\text{IS}^{(L)}| \leq 1$ . Indeed, as  $\alpha^{(L)}$  is a good dual solution of  $\text{DUAL}(L\epsilon_z) = \text{DUAL}(4n^7)$ , we claim  $H^{(L)}$  is a clique. First, note that any tight facility  $i$  has  $t_i \geq \frac{L\epsilon_z}{n} = 4n^6$  which means that all clients have a tight edge to  $i$  when  $i$  becomes tight (as the maximum squared facility-client distance is  $n^6$  by Lemma 4.1). Second, any two facilities  $i, i'$  have  $d(i, i')^2 \leq 4n^6$  using the triangle inequality and facility-client distance bound. Combining these two insights, we can see that  $H^{(L)}$  is a clique and so its independent set has size at most 1.

It remains to describe and analyze the procedure `QUASIGRAPHUPDATE` that will perform the interpolation between two conflict graphs  $H^{(\ell)}$  and  $H^{(\ell+1)}$  when given a maximal independent set  $\text{IS}^{(\ell)}$  of  $H^{(\ell)}$  so that  $|\text{IS}^{(\ell)}| > k$ . We run this procedure at most  $L$  times starting with  $H^{(0)}, H^{(1)}$ , and  $\text{IS}^{(0)} = \mathcal{F}$  until we find a solution of size  $k$ .

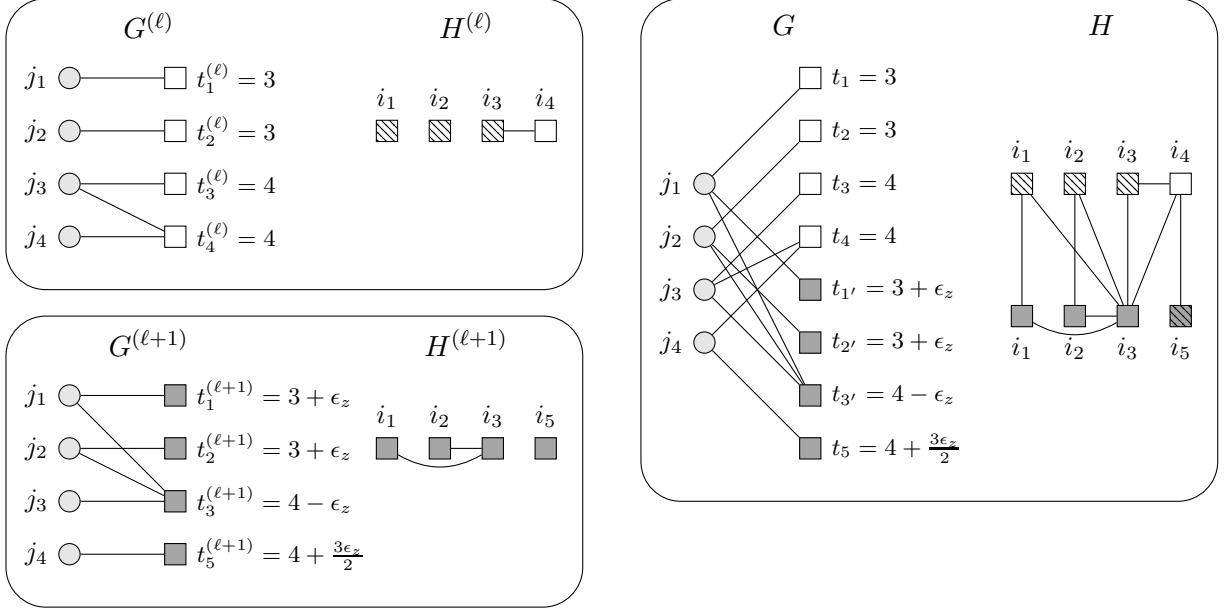


Figure 3: An example of the “hybrid” client-facility graph and associated conflict graph used by QUASIGRAPHUPDATE.  $G^{(\ell)}$  and  $G^{(\ell+1)}$  are the client-facility graphs of  $\alpha^{\text{in}}$  and  $\alpha^{\text{out}}$  of Figure 2. Next to the facilities, we have written the facility times ( $t_i$ ’s) of those solutions. As the squared-distance between any two facilities is 5 in the example of Figure 2, one can see that any two facilities with a common neighbor in the client-facility graph will be adjacent in the conflict graph.  $G$  is the “hybrid” client-facility graph of  $G^{(\ell)}$  and  $G^{(\ell+1)}$ . When  $H$  is formed, we extend the given maximal independent set  $\text{IS}^{(\ell)}$  of  $H^{(\ell)}$  to form a maximal independent set of  $H$ . The facilities in the relevant independent sets are indicated with stripes.

#### 4.2.1 Description of QUASIGRAPHUPDATE

Denote the input by  $H^{(\ell)}$ ,  $H^{(\ell+1)}$ , and  $\text{IS}^{(\ell)}$  (the maximal independent set of  $H^{(\ell)}$  of size greater than  $k$ ). Although we are interested in producing a sequence of conflict graphs, it will be helpful to think of a process that alters some “hybrid” client-facility graph  $G$ , then uses  $G$  and the corresponding opening times  $t$  to construct a new conflict graph  $H$  after each alteration. To ease the description of this process, we duplicate each facility that appears both in  $G^{(\ell)}$  and  $G^{(\ell+1)}$  so as to ensure that these sets are disjoint. Let  $\mathcal{V}^{(\ell)}$  and  $\mathcal{V}^{(\ell+1)}$  denote the (now disjoint) sets of facilities in  $G^{(\ell)}$  and  $G^{(\ell+1)}$ , respectively. Note that the duplication of facilities does not alter the solution space of the considered instance, as one may assume that at most one facility is opened at each location. Note that our algorithm will also satisfy this property, since  $d(i, i')^2 = 0$  for any pair of co-located facilities  $i, i'$ .

Initially, we let  $G$  be the client-facility graph with bipartition  $\mathcal{D}$  and  $\mathcal{V}^{(\ell)} \cup \mathcal{V}^{(\ell+1)}$  that has an edge from client  $j$  to facility  $i \in \mathcal{V}^{(\ell)}$  if  $(j, i)$  is present in  $G^{(\ell)}$  and to  $i \in \mathcal{V}^{(\ell+1)}$  if  $(j, i)$  is present in  $G^{(\ell+1)}$ . The opening time  $t_i$  of facility  $i$  is now naturally set to  $t_i^{(\ell)}$  if  $i \in \mathcal{V}^{(\ell)}$  and to  $t_i^{(\ell+1)}$  if  $i \in \mathcal{V}^{(\ell+1)}$ . Informally,  $G$  is the union of the two client-facility graphs  $G^{(\ell)}$  and  $G^{(\ell+1)}$  where the client vertices are shared (see Figure 3). We then generate<sup>2</sup> the conflict graph  $H^{(\ell,1)}$  from  $G$  and  $t$ . As the induced subgraph of  $H^{(\ell,1)}$  on vertex set  $\mathcal{V}^\ell$  equals  $H^{(\ell)} = H^{(\ell,0)}$ , we have that  $\text{IS}^{(\ell)}$  is

<sup>2</sup>Recall that a conflict graph is defined in terms of a client-facility graph  $G$  and  $t$ : the vertices are the facilities in  $G$ , and two facilities  $i$  and  $i'$  are adjacent if there is some client  $j$  that is adjacent to both of them in  $G$  and  $d(i, i')^2 \leq \delta \min(t_i, t_{i'})$ .

also an independent set of  $H^{(\ell,1)}$ . We obtain a maximal independent set  $\text{IS}^{(\ell,1)}$  of  $H^{(\ell,1)}$  by greedily extending  $\text{IS}^{(\ell)}$ . Clearly, the independent set can only increase so we still have  $|\text{IS}^{(\ell,1)}| > k$ .

To produce the remaining sequence, we iteratively perform changes, but construct and output a new conflict graph and maximal independent set after *each* such change. Specifically, we remove from  $G$  each facility  $i \in \mathcal{V}^{(\ell)}$ , one by one. At the end of the procedure (after  $|\mathcal{V}^{(\ell)}|$  many steps), we have  $G = G^{(\ell+1)}$  and so  $H^{(\ell,p_\ell)} = H^{(\ell+1)}$ . Note that at each step, our modification to  $G$  results in removing a single facility  $i$  from the associated conflict graph. Thus, if  $\text{IS}^{(\ell,s)}$  is an independent set in  $H^{(\ell,s)}$  before a modification, then  $\text{IS}^{(\ell,s)} \setminus \{i\}$  is an independent set in  $H^{(\ell,s+1)}$ . We obtain a maximal independent set  $\text{IS}^{(\ell,s+1)}$  of  $H^{(\ell,s+1)}$  by greedily extending  $\text{IS}^{(\ell,s)} \setminus \{i\}$ . Then, for each step  $s$ , we have  $|\text{IS}^{(\ell,s+1)}| \geq |\text{IS}^{(\ell,s)}| - 1$ , as required.

#### 4.2.2 Analysis

The total running time is  $n^{O(\epsilon^{-1} \log n)}$  since the number of steps  $L$  (and the number of dual solutions in our sequence) is  $n^{O(\epsilon^{-1} \log n)}$  and each step runs in polynomial time since it involves the construction of at most  $O(|\mathcal{F}|)$  conflict graphs and maximal independent sets.

We proceed to analyze the approximation guarantee. Consider the first time that we produce some maximal independent set  $\text{IS}$  of size exactly  $k$ . Suppose that when this happened, we were moving between two solutions  $\alpha^{(\ell)}$  and  $\alpha^{(\ell+1)}$ , i.e.,  $\text{IS} = \text{IS}^{(\ell,s)}$  is a maximal independent set of  $H^{(\ell,s)}$  for some  $1 \leq s \leq p_\ell$ . That we may assume that  $s \geq 1$  follows from  $|\text{IS}^{(0)}| > k$  and  $\text{IS}^{(\ell-1,p_\ell)} = \text{IS}^{(\ell)} = \text{IS}^{(\ell,0)}$  (recall that  $\text{IS}$  was selected to be the *first* independent set of size  $k$ ).

To ease notation, we let  $H = H^{(\ell,s)}$  and denote by  $G$  the “hybrid” client-facility graph that generated  $H$ . In order to analyze the cost of  $\text{IS}$ , let us form a hybrid solution  $\alpha$  by setting  $\alpha_j = \min(\alpha_j^{(\ell)}, \alpha_j^{(\ell+1)})$  for each client  $j \in \mathcal{D}$ . Note that  $\alpha \leq \alpha^{(\ell)}$  is a feasible solution of  $\text{DUAL}(\lambda)$  where  $\lambda = \ell \cdot \epsilon_z$  and, since  $\alpha^{(\ell)}$  and  $\alpha^{(\ell+1)}$  are close,  $\alpha_j \geq \alpha_j^{(\ell)} - \frac{1}{n^2}$  and  $\alpha_j \geq \alpha_j^{(\ell+1)} - \frac{1}{n^2}$ . For each client  $j$ , we define a set of facilities  $S_j \subseteq \text{IS}$  to which  $j$  contributes, as follows. For all  $i \in \text{IS}$ , we have  $i \in S_j$  if  $\alpha_j > d(j, i)^2$ . Note that  $S_j$  is a subset of  $j$ ’s neighborhood in  $G$  and therefore

$$\alpha_j = \min(\alpha_j^{(\ell)}, \alpha_j^{(\ell+1)}) \leq t_i = \begin{cases} t_i^{(\ell)} & \text{if } i \in \mathcal{V}^{(\ell)} \\ t_i^{(\ell+1)} & \text{if } i \in \mathcal{V}^{(\ell+1)} \end{cases} \quad \text{for all } i \in S_j.$$

Using the fact that  $\alpha^{(\ell+1)}$  is a good dual solution, we can bound the total service cost of all clients in the integral solution  $\text{IS}$ . Let us first proceed separately for those clients with  $|S_j| > 0$ . Let  $\mathcal{D}_0 = \{j \in \mathcal{D} : |S_j| = 0\}$ , and  $\mathcal{D}_{>0} = \mathcal{D} \setminus \mathcal{D}_0$ . We remark that the analysis is now very similar to the proof of Theorem 3.4. We define  $\beta_{ij} = [\alpha_j - d(i, j)^2]^+$  and similarly  $\beta_{ij}^{(\ell)} = [\alpha_j^{(\ell)} - d(i, j)^2]^+$  and  $\beta_{ij}^{(\ell+1)} = [\alpha_j^{(\ell+1)} - d(i, j)^2]^+$ .

**Lemma 4.6.** *For any  $j \in \mathcal{D}_{>0}$ ,  $d(j, \text{IS})^2 \leq \rho \cdot (\alpha_j - \sum_{i \in S_j} \beta_{ij})$ .*

*Proof.* Consider some  $j \in \mathcal{D}_{>0}$  and first suppose that  $|S_j| = 1$ . Then, if we let  $S_j = \{i\}$ ,  $\alpha_j = \beta_{ij} + d(j, i)^2 \geq \beta_{ij} + d(j, \text{IS})^2$  just as in “Case  $s = 1$ ” of Theorem 3.4. Next, suppose that  $|S_j| = s > 1$ . In other words,  $j$  is contributing to multiple facilities in  $\text{IS}$ . We then have  $\alpha_j - \sum_{i \in S_j} \beta_{ij} \geq \frac{1}{\rho} d(j, \text{IS})^2$  by the exact same arguments as in “Case  $s > 1$ ” of Theorem 3.4 using that we also in this case have  $\alpha_j \leq \min(t_i, t_{i'})$  for any two facilities  $i, i' \in S_j$ .  $\square$

Next, we bound the total service cost of all those clients that do not contribute to any facility in  $\text{IS}$ . The proof is very similar to “Case  $s = 0$ ” in the proof of Theorem 3.4.

**Lemma 4.7.** *For every  $j \in \mathcal{D}_0$ ,  $d(j, \text{IS})^2 \leq (1 + 5\epsilon)\rho \cdot \alpha_j$ .*

*Proof.* Consider some client  $j \in \mathcal{D}_0$ , and let  $i \in \mathcal{V}^{(\ell+1)}$  be a tight facility so that

$$(1 + \sqrt{\delta} + \epsilon)\sqrt{\alpha_j^{(\ell+1)}} \geq d(j, i) + \sqrt{\delta t_i^{(\ell+1)}}.$$

Such a facility  $i$  is guaranteed to exist because  $\alpha^{(\ell+1)}$  is a good dual solution. Furthermore, note that  $i$  is present in  $H$  since  $H$  contains all facilities in  $\mathcal{V}^{(\ell+1)}$ . By definition  $t_i = t_i^{(\ell+1)}$  and, as all  $\alpha$ -values are at least 1 (by the preprocessing of Lemma 4.1),  $(1 + \frac{1}{n^2})\alpha_j \geq \alpha_j + 1/n^2 \geq \alpha_j^{(\ell+1)}$ . Hence, the above inequality implies

$$(1 + \frac{1}{n^2})(1 + \sqrt{\delta} + \epsilon)\sqrt{\alpha_j} \geq d(j, i) + \sqrt{\delta t_i}.$$

Note the similarity of this inequality with that of (3.5) and the proof is now identical to “Case  $s = 0$ ” of Theorem 3.4.

Indeed, since  $\text{IS}$  is a maximal independent set of  $H$ , either  $i \in \text{IS}$ , in which case  $d(j, \text{IS}) \leq d(j, i)$ , or there is a  $i' \in \text{IS}$  such that the edge  $(i', i)$  is in  $H$ , in which case

$$d(j, \text{IS}) \leq d(j, i) + d(i, i') \leq d(j, i) + \sqrt{\delta t_i},$$

where the inequality follows from  $d(i, i')^2 \leq \delta \min(t_i, t_{i'})$  by the definition of  $H$ . In any case, we have (using  $n \gg 1/\epsilon$ )

$$d(j, \text{IS}) \leq (1 + \frac{1}{n^2})(1 + \sqrt{\delta} + \epsilon)\sqrt{\alpha_j} \leq (1 + 2\epsilon)(1 + \sqrt{\delta})\sqrt{\alpha_j}.$$

Squaring both sides and recalling that  $\rho \geq \frac{1}{(1+\sqrt{\delta})^2}$  and that  $\epsilon$  is a small constant so  $(1 + 2\epsilon)^2 \leq (1 + 5\epsilon)$  completes the proof of the lemma.  $\square$

One difference compared to the analysis in Section 3.2.2 is that not all opened facilities are fully paid for. However, they are almost paid for:

**Lemma 4.8.** *For any  $i \in \text{IS}$ ,  $\sum_{j \in \mathcal{D}} \beta_{ij} \geq \lambda - \frac{1}{n}$ .*

*Proof.* If  $i \in \mathcal{V}^{(\ell+1)}$ , then it is a tight facility with respect to  $\alpha^{(\ell+1)}$ , i.e.,  $\sum_{j \in \mathcal{D}} \beta_{ij}^{(\ell+1)} = \lambda + \epsilon_z$ . Similarly, if  $i \in \mathcal{V}^{(\ell)}$  then  $\sum_{j \in \mathcal{D}} \beta_{ij}^{(\ell)} = \lambda$ . Now since  $\alpha_j \geq \max(\alpha_j^{(\ell+1)}, \alpha_j^{(\ell)}) - \frac{1}{n^2}$  for every client  $j$ ,

$$\sum_{j \in \mathcal{D}} \beta_{ij} \geq \sum_{j \in \mathcal{D}} \left( \max(\beta_{ij}^{(\ell+1)}, \beta_{ij}^{(\ell)}) - \frac{1}{n^2} \right) \geq \lambda - \frac{1}{n}. \quad \square$$

We now combine the above lemmas to bound the approximation guarantee of the found solution. Recall that  $\text{OPT}_k$  denotes the optimum value of the standard LP-relaxation (see Section 2).

**Theorem 4.9.**  $\sum_{j \in \mathcal{D}} d(j, \text{IS})^2 \leq (1 + 6\epsilon)\rho \cdot \text{OPT}_k$ .

*Proof.* From Lemmas 4.6 and 4.7 we have:

$$\sum_{j \in \mathcal{D}} d(j, \text{IS})^2 \leq (1 + 5\epsilon)\rho \sum_{j \in \mathcal{D}} \left( \alpha_j - \sum_{i \in S_j} \beta_{ij} \right).$$

By Lemma 4.8 (note that by definition,  $\sum_{i \in \mathcal{IS}} \beta_{ij} = \sum_{i \in S_j} \beta_{ij}$ ),

$$\sum_{j \in \mathcal{D}} \left( \alpha_j - \sum_{i \in S_j} \beta_{ij} \right) \leq \sum_{j \in \mathcal{D}} \alpha_j - |\mathcal{IS}| \left( \lambda - \frac{1}{n} \right) = \sum_{j \in \mathcal{D}} \alpha_j - k \cdot \lambda + \frac{k}{n} \leq \text{OPT}_k + 1,$$

where the last inequality follows from  $k \leq n$  and, as  $\alpha$  is a feasible solution to  $\text{DUAL}(\lambda)$ ,  $\sum_{j \in \mathcal{D}} \alpha_j - k \cdot \lambda \leq \text{OPT}_k$ . The statement now follows from  $\text{OPT}_k \geq \sum_{j \in \mathcal{D}} \min_{i \in \mathcal{F}} d(i, j)^2 \geq n$  and  $n \gg 1/\epsilon$ .  $\square$

We have thus proved that our quasi-polynomial algorithm produces a  $(\rho + O(\epsilon))$ -approximate solution which implies Theorem 1.1. The quasi-polynomial algorithms for the other considered problems are the same except for the selection of  $\delta$  and  $\rho$ , and that in the  $k$ -median problem the connection costs are the (non-squared) distances.

## 5 A Polynomial Time Approximation Algorithm

We now show how to obtain a polynomial-time algorithm, building on the ideas presented in the previous section. As in Section 4, we focus exclusively on the  $k$ -means problem, and let  $\delta = \delta_{\text{mean}} \approx 2.3146$  and  $\rho = \rho_{\text{mean}} = (1 + \sqrt{\delta})^2 \approx 6.3574$ , and assume that the squared-distances between clients and facilities are in  $[1, n^6]$  by Lemma 4.1. Additionally, we choose  $\epsilon$  and  $\gamma$  to be suitably small constants with  $0 < \gamma \ll \epsilon \ll 1$ , and for notational convenience we assume without loss of generality that  $n \gg 1/\gamma$ .

Similarly to Section 4.1, we give an algorithm for generating a close sequence of feasible solutions to  $\text{DUAL}(\lambda)$ , and then show how to use this sequence to generate a sequence of integral solutions that must contain some solution of size exactly  $k$ . Here, however, we ensure that our sequence of feasible solutions is of polynomial length. In order to accomplish this, we must relax some of the requirements in our definition of a good solution (Definition 4.2).

First, rather than requiring that every facility have opening cost  $\lambda$ , we instead allow each facility  $i$  to have its own price in  $z_i \in \{\lambda, \lambda + \frac{1}{n}\}$  (Condition 1 of Definition 5.1). For each  $\alpha^{(\ell)}$ , our algorithm will produce an associated set of facility prices  $z^{(\ell)} = \{z_i^{(\ell)}\}_{i \in \mathcal{F}}$ . For any such  $(\alpha^{(\ell)}, z^{(\ell)})$ , we define  $\beta_{ij}^{(\ell)} = [\alpha_j^{(\ell)} - d(j, i)^2]^+$  and  $N^{(\ell)}(i) = \{j : \beta_{ij}^{(\ell)} > 0\}$ , as before. However, we now say that a facility  $i$  is *tight* in  $(\alpha^{(\ell)}, z^{(\ell)})$  if  $\sum_{j \in \mathcal{D}} \beta_{ij}^{(\ell)} = z_i^{(\ell)}$ . That is, we consider a facility  $i$  tight once its (possibly unique) price  $z_i$  is paid in the dual. Intuitively, if all the facility prices  $z_i$  are *almost* the same, we can still carry out our analysis, and obtain a  $(\rho + O(\epsilon))$ -approximation.

Second, we shall designate a set of *special* facilities  $\mathcal{F}_s \subseteq \mathcal{F}$  that we shall open, *even if they are not tight*. To each special facility  $i \in \mathcal{F}_s$  we assign a set of special clients  $\mathcal{D}_s(i) \subseteq \mathcal{D}$  that are allowed to pay for  $i$ . Then, for each  $i \in \mathcal{F}_s$ , we define the time  $\tau_i = \max_{j \in N(i) \cap \mathcal{D}_s(i)} \alpha_j$ , while for each  $i \in \mathcal{F} \setminus \mathcal{F}_s$  we set  $\tau_i = t_i = \max_{j \in N(i)} \alpha_j$ . Again, we adopt the convention that  $\tau_i = 0$  if  $N(i) \cap \mathcal{D}_s(i) = \emptyset$  for  $i \in \mathcal{F}_s$  or  $N(i) = \emptyset$  for  $i \in \mathcal{F} \setminus \mathcal{F}_s$ . Although a facility in  $\mathcal{F}_s$  is not necessarily tight, we shall require that the total of all payments to such facilities by special clients is *almost* equal to  $\lambda|\mathcal{F}_s|$  (Condition 3 of Definition 5.1). That is, on average, each facility of  $\mathcal{F}_s$  is almost tight.

Finally, given the times  $\tau_i$ , we shall not require that *every* client  $j$  have some tight or special facility  $i$  such that  $(1 + \sqrt{\delta} + 10\epsilon)\sqrt{\alpha_j} \geq d(j, i) + \sqrt{\delta\tau_i}$ . Specifically, we shall allow some small set of *bad* clients  $\mathcal{D}_b$  to instead satisfy a weaker inequality  $6\sqrt{\alpha_j} \geq d(j, i) + \sqrt{\delta\tau_i}$  for some tight or special facility  $i$ . Such clients will have a higher service cost, so we require that their total contribution to the cost of an optimal solution is small (Condition 2 of Definition 5.1).

Combining the above, we have the following definition.

**Definition 5.1.** Consider a tuple  $(\alpha, z, \mathcal{F}_S, \mathcal{D}_S)$  where  $\alpha \in \mathbb{R}^{\mathcal{D}}, z \in \mathbb{R}^{\mathcal{F}}, \mathcal{F}_S \subseteq \mathcal{F}$  is a set of special facilities, and  $\mathcal{D}_S : \mathcal{F}_S \rightarrow 2^{\mathcal{D}}$  is a function assigning each special facility  $i$  a set of special clients  $\mathcal{D}_S(i)$ . We say that this tuple is roundable for  $\lambda$  (or  $\lambda$ -roundable) if  $\alpha$  is a feasible solution of  $\text{DUAL}(\lambda + \frac{1}{n})$ , and:

1. For all  $i \in \mathcal{F}$ ,  $\lambda \leq z_i \leq \lambda + \frac{1}{n}$ .
2. There exists a subset  $\mathcal{D}_B$  of clients so that for all  $j \in \mathcal{D}$  there is a facility  $w(j)$  that is either tight or in  $\mathcal{F}_S$  and:

$$(a) \quad (1 + \sqrt{\delta} + 10\epsilon)^2 \alpha_j \geq \left( d(j, w(j)) + \sqrt{\delta \cdot \tau_{w(j)}} \right)^2 \text{ for all } j \in \mathcal{D} \setminus \mathcal{D}_B.$$

$$(b) \quad 36\gamma \cdot \text{OPT}_k \geq \sum_{j \in \mathcal{D}_B} \left( d(j, w(j)) + \sqrt{\delta \cdot \tau_{w(j)}} \right)^2,$$

3.  $\sum_{i \in \mathcal{F}_S} \sum_{j \in \mathcal{D}_S(i)} \beta_{ij} \geq \lambda |\mathcal{F}_S| - \gamma \cdot \text{OPT}_k$  and  $|\mathcal{F}_S| \leq n$ .

Observe that any  $\lambda$ -roundable solution with  $\mathcal{F}_S = \emptyset$ , and  $\mathcal{D}_B = \emptyset$  is essentially a good solution for  $\text{DUAL}(\lambda + \frac{1}{n})$  (as defined for the quasi-polynomial algorithm in Section 4) except that the opening costs of the facilities are allowed to vary slightly. We shall also say that  $(\alpha, z)$  is roundable if  $(\alpha, z, \emptyset, \mathcal{D}_S)$  is roundable.

An overview of our polynomial time algorithm is shown in Algorithm 1. The algorithm maintains a current base price  $\lambda$  and a current roundable solution  $\mathcal{S}^{(0)} = (\alpha^{(0)}, z^{(0)}, \mathcal{F}_S^{(0)}, \mathcal{D}_S^{(0)})$  for  $\lambda$ , as well as a corresponding integral solution  $\text{IS}^{(0)}$ . As in the quasi-polynomial algorithm, we shall enumerate a sequence  $0, 1 \cdot \epsilon_z, 2 \cdot \epsilon_z, \dots, L \cdot \epsilon_z$  of base prices  $\lambda$ , where now  $\epsilon_z = n^{-O(1)}$  and, as before we define  $L = 4n^7 \cdot \epsilon_z^{-1}$ . Here, however we increase facility prices from  $\lambda$  to  $\lambda + \epsilon_z$  *one-by-one* using an auxiliary procedure **RAISEPRICE**, which takes as input a fractional dual solution  $\alpha^{(0)}$ , a set of prices  $z^{(0)}$ , a current integral solution  $\text{IS}^{(0)}$ , and a facility  $i$ . **RAISEPRICE** increases the price of facility  $i$ , then outputs a close sequence of roundable solutions  $\mathcal{S}^{(1)} = (\alpha^{(1)}, z^{(1)}, \mathcal{F}_S^{(1)}, \mathcal{D}_S^{(1)}), \dots, \mathcal{S}^{(q)} = (\alpha^{(q)}, z^{(q)}, \mathcal{F}_S^{(q)}, \mathcal{D}_S^{(q)})$ , each having  $z_i^{(\ell)} = z_i^{(0)} + \epsilon_z$  and  $z_{i'}^{(\ell)} = z_{i'}^{(0)}$  for all  $i' \neq i$ . Note that in addition to increasing the facility prices one-by-one, we now generate a *sequence* of solutions for each individual price increase.

Initially, we set  $\lambda \leftarrow 0$  and then initialize  $\mathcal{S}^{(0)}$  by setting  $z_i^{(0)} \leftarrow 0$  for all  $i \in \mathcal{F}$  and  $\mathcal{F}_S = \emptyset$  (observe that  $\mathcal{D}_S$  is then an empty function), and constructing  $\alpha^{(0)}$  as follows. We set  $\alpha_j = 0$  for all  $j \in \mathcal{D}$  and then increase all  $\alpha_j$  at a uniform rate. We stop increasing a value  $\alpha_j$  whenever  $j$  gains a tight edge to some facility  $i \in \mathcal{F}$  or  $2\sqrt{\alpha_j} \geq d(j, j') + 6\sqrt{\alpha_{j'}}$  for some  $j' \in \mathcal{D}$  (the rationale behind this choice will be made clear in Section 7). Finally, we initialize our current integral solution  $\text{IS}^{(0)} = \mathcal{F}$ .

As long as an integral solution of size  $k$  has not yet been produced, Algorithm 1 iterates through each facility  $i \in \mathcal{F}$ , calling **RAISEPRICE** to raise  $z_i$  by  $\epsilon_z < 1/n$ . The sequences that are produced are used to obtain a sequence of integral solutions in which the size of each solution decreases by at most 1. This is done by using a second procedure, **GRAPHUPDATE**, which is very similar to the procedure **QUASIGRAPHUPDATE** described in the previous section. Note that raise price always increases a single facility  $i$ 's price by  $\epsilon_z < 1/n$ , and does not increase  $z_i$  further until all other facility prices have also been increased by  $\epsilon_z$ . Thus, each in every pair of consecutive solutions  $\mathcal{S}^{(\ell)}, \mathcal{S}^{(\ell+1)}$  considered by **GRAPHUPDATE** in line 7, every price  $z_i \in \{\lambda, \lambda + \epsilon_z\}$  and so both solutions are  $\lambda$ -roundable (for the same value  $\lambda$ ). We describe our auxiliary procedures **GRAPHUPDATE** and **RAISEPRICE** in the next sections. Note that initially  $|\text{IS}^{(0)}| = |\mathcal{F}|$  and, by the same reasoning as in

Section 4.2, once  $\lambda = L \cdot \epsilon_z = 4n^7$  we must have  $|\text{IS}^{(0)}| = 1$ . Thus, at some intermediate point, we will indeed find some solution  $\text{IS}$  of size  $k$ .

---

**Algorithm 1:** Polynomial time  $(\rho_{\text{mean}} + O(\epsilon))$ -approximation algorithm for  $k$ -means

---

```

1 Initialize  $\mathcal{S}^{(0)} = (\alpha^{(0)}, z^{(0)}, \mathcal{F}_S^{(0)}, \mathcal{D}_S^{(0)})$  as described in our discussion above
2  $\lambda \leftarrow 0, \text{IS}^{(0)} \leftarrow \mathcal{F}$ 
3 for  $\lambda = 0, 1 \cdot \epsilon_z, 2 \cdot \epsilon_z, \dots, L \cdot \epsilon_z$  do
    /* Raise the price of each facility  $i$  to  $z_i^{(0)} + \epsilon_z = \lambda + \epsilon_z$  */
4   foreach  $i \in \mathcal{F}$  do
5     Call RAISEPRICE( $\alpha^{(0)}, z^{(0)}, \text{IS}^{(0)}, i$ ) to produce a sequence  $\mathcal{S}^{(1)}, \dots, \mathcal{S}^{(q)}$  of  $\lambda$ -roundable
       solutions
       /* Move through this sequence, constructing integral solutions */
6     for  $\ell = 0$  to  $q - 1$  do
7       Call GRAPHUPDATE( $\mathcal{S}^{(\ell)}, \mathcal{S}^{(\ell+1)}, \text{IS}^{(\ell)}$ ) to produce a sequence  $\text{IS}^{(\ell,0)}, \dots, \text{IS}^{(\ell,p_\ell)}$ 
8       if  $|\text{IS}^{(\ell,r)}| = k$  for some  $\text{IS}^{(\ell,r)}$  in this sequence then return  $\text{IS}^{(\ell,r)}$  else  $\text{IS}^{(\ell+1)} \leftarrow \text{IS}^{(\ell,p_\ell)}$ 
       /* After each price increase, update current solutions */
9      $\mathcal{S}^{(0)} \leftarrow \mathcal{S}^{(q)}, \text{IS}^{(0)} \leftarrow \text{IS}^{(q)}$ 
    /* All prices have been increased. Continue to the next base price  $\lambda$  */
```

---

Algorithm 1 executes  $L = 4n^7 \cdot \epsilon_z^{-1}$  base price increases, each of which performs  $|\mathcal{F}|$  calls to RAISEPRICE. In order to show that Algorithm 1 runs in polynomial time, it is sufficient to show that each call to RAISEPRICE and GRAPHUPDATE produces a polynomial length sequence in polynomial time. In the next sections, we describe these procedures in more detail and show that they run in polynomial time. In addition, we show that RAISEPRICE produces a sequence of roundable solutions (Proposition 8.18) that are close (Proposition 8.10). In Section 6, we show that given these solutions, GRAPHUPDATE finds a  $(\rho + 1000\epsilon)$ -approximate solution (Theorem 6.4).<sup>3</sup> This implies our main theorem:

**Theorem 5.2.** *For any  $\epsilon > 0$ , there is a  $(\rho + \epsilon)$ -approximation algorithm for  $k$ -means.*

## 6 Opening a Set of Exactly $k$ Facilities in a Close, Roundable Sequence

In this section, we describe our algorithm GRAPHUPDATE for interpolating between two close roundable solutions  $\mathcal{S}^{(\ell)}$  and  $\mathcal{S}^{(\ell+1)}$  starting with a maximal independent set  $\text{IS}^{(\ell)}$  of the conflict graph<sup>4</sup>  $H^{(\ell)}$  of  $\mathcal{S}^{(\ell)}$ . The goal of this procedure is the same as that of QUASIGRAPHUPDATE explained in Section 4.2: we maintain a sequence of maximal independent sets in appropriately constructed conflict graphs so that the size of the independent set never decreases by more than 1, and the last solution is a maximal independent set of the conflict graph  $H^{(\ell+1)}$  of  $\mathcal{S}^{(\ell+1)}$ . Similar to Section 4.2, we use a “hybrid” client-facility graph to generate our conflict graph in each step of our procedure. The only difference is that we need to slightly generalize the definition of a client-facility graph to incorporate the concept of roundable solutions.

<sup>3</sup>We remark that we have chosen to first describe GRAPHUPDATE as that procedure is very similar to QUASIGRAPHUPDATE in the quasi-polynomial algorithm whereas RAISEPRICES is more complex.

<sup>4</sup>Below, we slightly generalize the definition of client-facility and conflict graphs in Section 3.1 to that of roundable solutions.



**Client-facility and conflict graphs of roundable solutions.** We define the *client-facility* graph  $G$  of a roundable solution  $\mathcal{S} = (\alpha, z, \mathcal{F}_s, \mathcal{D}_s)$  as in Section 3.1 with the following two changes: First, recall that we now consider a facility  $i$  *tight* if and only if  $\sum_{j \in N(i)} \beta_{ij} = z_i$ . Second, we shall additionally add every facility  $i \in \mathcal{F}_s$  to  $G$ , but place an edge between each  $i \in \mathcal{F}_s$  and  $j \in \mathcal{D}$  only if  $j \in N(i) \cap \mathcal{D}_s(i)$ . Intuitively, we treat special facilities  $i \in \mathcal{F}_s$  essentially the same as tight facilities, except only those clients in  $N(i) \cap \mathcal{D}_s(i)$  are considered to be paying for  $i$ .

Formally, let  $\mathcal{V}$  denote the set of all tight facilities or special facilities with respect to  $\mathcal{S}$ . Then,  $G$  is a bipartite graph on  $\mathcal{D}$  and  $\mathcal{V}$  that contains an edge  $(i, j)$  if and only if  $i \in \mathcal{V} \setminus \mathcal{F}_s$  and  $j \in N(i)$  or  $i \in \mathcal{F}_s$  and  $j \in N(i) \cap \mathcal{D}_s(i)$ . As before, we assign an *opening time*  $\tau_i$  to each  $i \in \mathcal{V}$ . For  $i \in \mathcal{V} \setminus \mathcal{F}_s$ ,  $\tau_i = t_i = \max_{j \in N(i)} \alpha_j$ , and for  $i \in \mathcal{F}_s$ ,  $\tau_i = \max_{j \in N(i) \cap \mathcal{D}_s(i)} \alpha_j$ . In other words,  $\tau_i$  equals the maximum  $\alpha_j$  over all clients  $j$  such that  $(j, i)$  is an edge in  $G$  (in the case that there is no such edge, we adopt the convention that  $\tau_i = 0$ ).

Given a client facility graph  $G$ , and a set of opening times  $\tau$ , we construct the corresponding *conflict graph*  $H$  in the same way as in Section 3: the vertex set of  $H$  is the set of all facilities appearing in  $G$  and we place an edge between two facilities  $i$  and  $i'$  in  $H$  if and only if there is some  $j \in \mathcal{D}$  such that both  $(j, i)$  and  $(j, i')$  are present in  $G$  and  $d(i, i')^2 \leq \delta \cdot \min(\tau_i, \tau_{i'})$ . Notice that this coincides with the definition in Section 3 when the set of special facilities is empty. In particular, the initial independent set  $\mathcal{F}$  is a maximal independent set of the conflict graph associated to the initial solution (which has all facilities and no edges). Then, as in each iteration the last constructed independent set by GRAPHUPDATE is given as input in the next call (see Algorithm 1), we maintain the property that the input independent set  $\text{IS}^{(\ell)}$  is a maximal independent set of the conflict graph of  $\mathcal{S}^{(\ell)}$ .

**Description of GRAPHUPDATE.** Our algorithm now proceeds in the exact same way as QUASI-GRAPHUPDATE in Section 4.2. A short description is repeated here for convenience. Let  $G^{(\ell)}, \tau^{(\ell)}$  and  $G^{(\ell+1)}, \tau^{(\ell+1)}$  be the client-facility graphs and times associated with  $\mathcal{S}^{(\ell)}$  and  $\mathcal{S}^{(\ell+1)}$ , respectively. Furthermore, let  $H^{(\ell)}$  and  $H^{(\ell+1)}$  be the conflict graphs generated by  $G^{(\ell)}, \tau^{(\ell)}$  and  $G^{(\ell+1)}, \tau^{(\ell+1)}$ . Recall that the input to GRAPHUPDATE is  $\mathcal{S}^{(\ell)}, \mathcal{S}^{(\ell+1)}$  and a maximal independent set  $\text{IS}^{(\ell)}$  of  $H^{(\ell)}$ .

Define the “hybrid” client-facility graph  $G$  as the union of  $G^{(\ell)}$  and  $G^{(\ell+1)}$  where the client vertices are shared and the facilities are duplicated if necessary so as to make sure that the facilities of  $G^{(\ell)}$  and  $G^{(\ell+1)}$  are disjoint. The opening times are defined by

$$\tau_i = \begin{cases} \tau_i^{(\ell)} & \text{if } i \in \mathcal{V}^{(\ell)} \\ \tau_i^{(\ell+1)} & \text{if } i \in \mathcal{V}^{(\ell+1)} \end{cases},$$

where  $\mathcal{V}^{(\ell)}$  and  $\mathcal{V}^{(\ell+1)}$  denote the (disjoint) sets of facilities in  $G^{(\ell)}$  and  $G^{(\ell+1)}$ , respectively. We then generate the conflict graph  $H^{(\ell,1)}$  from  $G$  and  $\tau$ . As the induced subgraph of  $H^{(\ell,1)}$  on vertex set  $\mathcal{V}^{(\ell)}$  equals  $H^{(\ell)} = H^{(\ell,0)}$ , we have that the given maximal independent set  $\text{IS}^{(\ell)}$  of  $H^{(\ell)}$  is also an independent set of  $H^{(\ell,1)}$ . We obtain a maximal independent set  $\text{IS}^{(\ell,1)}$  of  $H^{(\ell,1)}$  by greedily extending  $\text{IS}^{(\ell)}$ . We then obtain the remaining conflict graphs and independent sets by removing from  $G$  each facility  $i \in \mathcal{V}^{(\ell)}$ , one by one. After each step we generate the associated conflict graph and we greedily extend the previous independent set (with  $i$  potentially removed) so as to obtain a maximal independent set in the updated conflict graph. This results, as in Section 4.2, in the sequence  $H^{(\ell)} = H^{(\ell,0)}, H^{(\ell,1)}, \dots, H^{(\ell,p_\ell)} = H^{(\ell+1)}$  of  $|\mathcal{V}^{(\ell)}| + 2$  many conflict graphs and a sequence  $\text{IS}^{(\ell)} = \text{IS}^{(\ell,0)}, \text{IS}^{(\ell,1)}, \dots, \text{IS}^{(\ell,p_\ell)} = \text{IS}^{(\ell+1)}$  of associated maximal independent sets so that  $|\text{IS}^{(\ell,s)}| \geq |\text{IS}^{(\ell,s-1)}| - 1$  for any  $s = 1, \dots, p_\ell$ . The output of GRAPHUPDATE is this sequence of independent sets.

## 6.1 Analysis

GRAPHUPDATE clearly runs in polynomial time since the number of steps is polynomial and each step requires only the construction of a conflict graph and greedily maintaining a maximal independent set.

We proceed to analyze the approximation guarantee. In comparison to Section 4.2.2, our analysis here is slightly more involved because it is with respect to roundable solutions instead of good solutions. In addition, we prove that *all* independent sets constructed in Algorithm 1 (by calls to GRAPHUPDATE) of size at least  $k$  have small connection cost. Specifically, we show that any constructed independent set  $\text{IS}$  with  $|\text{IS}| \geq k$  has  $\sum_{j \in \mathcal{D}} d(j, \text{IS})^2 \leq (\rho + O(\epsilon)) \text{OPT}_k$ .

First note that the initial independent set  $\text{IS}^{(0)}$  of Algorithm 1 contains all facilities and hence  $\sum_{j \in \mathcal{D}} d(j, \text{IS}^{(0)})^2 \leq \text{OPT}_k$ . All other independent sets are constructed by calls to GRAPHUPDATE. Consider one such independent set  $\text{IS}$  with  $|\text{IS}| \geq k$  and consider the first time this independent set was constructed. Suppose that when this happened, we were moving between two solutions  $\mathcal{S}^{(\ell)}$  and  $\mathcal{S}^{(\ell+1)}$  that are roundable for the same  $\lambda$ . Then,  $\text{IS} = \text{IS}^{(\ell, s)}$  for some step  $s \geq 1$  of GRAPHUPDATE. We may assume  $s \geq 1$  because  $\text{IS}^{(\ell, 0)} = \text{IS}^{(\ell)}$  was constructed in the previous call to GRAPHUPDATE (or it equals the initial independent set). Let  $G$  and  $\tau$  be the client-facility graph and the opening times that generated the conflict graph  $H = H^{(\ell, s)}$  in which  $\text{IS} = \text{IS}^{(\ell, s)}$  is a maximal independent set. Also note that we may assume, without loss of generality, that  $|\text{IS}| \leq n$ . Otherwise, we can reduce the size of  $\text{IS}$  since the connection cost of  $\text{IS}$  equals that of  $\bigcup_{j \in \mathcal{D}} \{\arg \min_{i \in \text{IS}} d(j, i)\}$ .

Similar to Section 4.2.2, we analyze the cost of  $\text{IS}$  with respect to a hybrid solution  $\alpha$  obtained by setting  $\alpha_j = \min(\alpha_j^{(\ell)}, \alpha_j^{(\ell+1)})$  for each client  $j \in \mathcal{D}$ . The following observations and concepts are also very similar to the ones in that section. We remark that  $\alpha$  is a feasible solution of  $\text{DUAL}(\lambda + \frac{1}{n})$  and, since  $\alpha^{(\ell)}$  and  $\alpha^{(\ell+1)}$  are close,  $\alpha_j \geq \alpha_j^{(\ell)} - \frac{1}{n^2}$  and  $\alpha_j \geq \alpha_j^{(\ell+1)} - \frac{1}{n^2}$ . For each client  $j$ , we define a set of facilities  $S_j \subseteq \text{IS}$  to which  $j$  contributes, as follows. For all  $i \in \text{IS}$ , we have  $i \in S_j$  if  $\alpha_j > d(j, i)^2$  and  $(j, i)$  is an edge in  $G$ . Note that  $S_j$  is a subset of  $j$ 's neighborhood in  $G$  and therefore

$$\alpha_j = \min(\alpha_j^{(\ell)}, \alpha_j^{(\ell+1)}) \leq \tau_i \quad \text{for all } i \in S_j. \quad (6.1)$$

Using the fact that  $\mathcal{S}^{(\ell+1)}$  is roundable, we can bound the total service cost of all clients in the integral solution  $\text{IS}$ . Let us first proceed separately for those clients with  $|S_j| > 0$ . Let  $\mathcal{D}_0 = \{j \in \mathcal{D} : |S_j| = 0\}$ , and  $\mathcal{D}_{>0} = \mathcal{D} \setminus \mathcal{D}_0$ . The following lemma is identical to Lemma 4.6 and its proof is therefore omitted.

**Lemma 6.1.** *For any  $j \in \mathcal{D}_{>0}$ ,  $d(j, \text{IS})^2 \leq \rho \cdot \left(\alpha_j - \sum_{i \in S_j} \beta_{ij}\right)$ .*

Next, we bound the total service cost of all those clients that do not contribute to any facility in  $\text{IS}$ . The proof is very similar to that of Lemma 4.7 except that we also need to handle the bad clients in  $\mathcal{D}_B$ .

**Lemma 6.2.**  $\sum_{j \in \mathcal{D}_0} d(j, \text{IS})^2 \leq (\rho + 200\epsilon) \sum_{j \in \mathcal{D}_0} \alpha_j + 36\gamma \cdot \text{OPT}_k$ .

*Proof.* Consider some client  $j \in \mathcal{D}_0$ , and let  $w(j) \in \mathcal{V}^{(\ell+1)}$  be the tight or special facility for  $j$  corresponding to the roundable solution  $\mathcal{S}^{(\ell+1)}$ . Note that  $w(j)$  is present in  $H$  (since  $H = H^{(\ell, s)}$  with  $s \geq 1$  contains all facilities in  $\mathcal{V}^{(\ell+1)}$ ) and  $\tau_{w(j)} = \tau_{w(j)}^{(\ell+1)}$  by definition. Thus, since  $\text{IS}$  is a maximal independent set of  $H$ , either  $w(j) \in \text{IS}$ , in which case  $d(j, \text{IS}) \leq d(j, w(j))$ , or there must be some other facility  $i \in \text{IS}$  such that  $H$  contains the edge  $(i, w(j))$ , in which case

$$d(j, \text{IS}) \leq d(j, w(j)) + d(w(j), i) \leq d(j, w(j)) + \sqrt{\delta \tau_i},$$

where the inequality follows from the fact that  $i$  and  $w(j)$  are adjacent in  $H$  and thus  $d(i, w(j))^2 \leq \delta \min(\tau_i, \tau_{i'})$  by the definition of  $H$ . In any case, we have  $d(j, \text{IS}) \leq d(j, w(j)) + \sqrt{\delta \tau_i}$  with  $\tau_i = \tau_i^{(\ell+1)}$ , and so:

$$\begin{aligned} \sum_{j \in \mathcal{D}_0} d(j, \text{IS})^2 &= \sum_{j \in \mathcal{D}_0 \setminus \mathcal{D}_B} d(j, \text{IS})^2 + \sum_{j \in \mathcal{D}_0 \cap \mathcal{D}_B} d(j, \text{IS})^2 \\ &\leq \sum_{j \in \mathcal{D}_0 \setminus \mathcal{D}_B} \left( d(j, w(j)) + \sqrt{\delta \cdot \tau_{w(j)}^{(\ell+1)}} \right)^2 + \sum_{j \in \mathcal{D}_B} \left( d(j, w(j)) + \sqrt{\delta \cdot \tau_{w(j)}^{(\ell+1)}} \right)^2 \\ &\leq \sum_{j \in \mathcal{D}_0 \setminus \mathcal{D}_B} (1 + \sqrt{\delta} + 10\epsilon)^2 \cdot \alpha_j^{(\ell+1)} + 36\gamma \cdot \text{OPT}_k, \end{aligned}$$

where the final inequality follows from the fact that  $\mathcal{S}^{(\ell+1)}$  is roundable. The statement now follows since  $\alpha_j^{(\ell+1)} \leq \alpha_j + 1/n^2 \leq (1 + 1/n^2)\alpha_j$  for all  $j \in \mathcal{D}$ ,  $\epsilon \leq 1$ ,  $\sqrt{\delta} \leq 2$ , and  $\rho \geq (1 + \sqrt{\delta})^2$ .  $\square$

We now bound the contributions to the opened facilities as in Lemma 4.8 except that we also need to handle the special facilities.

**Lemma 6.3.** *For any  $i \in \text{IS} \setminus (\mathcal{F}_S^{(\ell)} \cup \mathcal{F}_S^{(\ell+1)})$ , we have  $\sum_{j \in \mathcal{D}} \beta_{ij} \geq \lambda - \frac{1}{n}$  and for any  $i \in \mathcal{F}_S^{(x)}$  for some  $x \in \{\ell, \ell+1\}$ , we have  $\sum_{j \in \mathcal{D}_S^{(x)}(i)} \beta_{ij} \geq \left( \sum_{j \in \mathcal{D}_S^{(x)}(i)} \beta_{ij}^{(x)} \right) - \frac{1}{n}$ .*

*Proof.* For the first bound, consider a facility  $i \in \text{IS} \setminus (\mathcal{F}_S^{(\ell)} \cup \mathcal{F}_S^{(\ell+1)})$  and let  $x \in \{\ell, \ell+1\}$  be such that  $i \in \mathcal{V}^{(x)}$ . Then  $i$  is a tight facility with respect to  $(\alpha^{(x)}, z^{(x)})$ , i.e.,  $\sum_{j \in \mathcal{D}} \beta_{ij}^{(x)} = z_i^{(x)}$ . As  $\mathcal{S}^{(x)}$  is roundable for  $\lambda$ , we have  $z_i^{(x)} \geq \lambda$ . Moreover,  $\alpha_j \geq \alpha_j^{(x)} - \frac{1}{n^2}$  for every client  $j$ , and so

$$\sum_{j \in \mathcal{D}} \beta_{ij} \geq \sum_{j \in \mathcal{D}} \left( \beta_{ij}^{(x)} - \frac{1}{n^2} \right) \geq \lambda - \frac{1}{n}.$$

Now consider a special facility  $i \in \mathcal{F}_S^{(x)}$  for some  $x \in \{\ell, \ell+1\}$ . Then, by again using that  $\alpha_j \geq \alpha_j^{(x)} - \frac{1}{n^2}$  for every client  $j$ ,

$$\sum_{j \in \mathcal{D}_S^{(x)}(i)} \beta_{ij} \geq \sum_{j \in \mathcal{D}_S^{(x)}(i)} \left( \beta_{ij}^{(x)} - \frac{1}{n^2} \right),$$

and the lemma follows since  $|\mathcal{D}_S^{(x)}(i)| \leq |\mathcal{D}| = n$ .  $\square$

We are now ready to prove our main result, which bounds the connection cost of  $\text{IS}$  in terms of  $\text{OPT}_k$  as desired. The proof is very similar to the proof of Theorem 4.9.

**Theorem 6.4.** *For any  $\text{IS}$  produced by  $\text{GRAPHUPDATE}$  with  $|\text{IS}| \geq k$ ,*

$$\sum_{j \in \mathcal{D}} d(j, \text{IS})^2 \leq (\rho + 1000\epsilon) \cdot \text{OPT}_k.$$

*Proof.* From Lemmas 6.1 and 6.2 we have:

$$\sum_{j \in \mathcal{D}} d(j, \text{IS})^2 \leq (\rho + 200\epsilon) \left( \sum_{j \in \mathcal{D}} \alpha_j - \sum_{i \in S_j} \beta_{ij} \right) + 36\gamma \cdot \text{OPT}_k. \quad (6.2)$$

Note that by definition, if  $i \notin \mathcal{F}_S^{(\ell)} \cup \mathcal{F}_S^{(\ell+1)}$  then  $\sum_{j \in \mathcal{D}} \beta_{ij} = \sum_{j: i \in S_j} \beta_{ij}$  and if  $i \in \mathcal{F}_S^{(x)}$  then  $\sum_{j \in \mathcal{D}_S^{(x)}(i)} \beta_{ij} = \sum_{j: i \in S_j} \beta_{ij}$ . Also, recall that by our construction of  $H$ ,  $\mathcal{F}_S^{(\ell)}$  and  $\mathcal{F}_S^{(\ell+1)}$  are distinct. Thus, by Lemma 6.3,

$$\begin{aligned} \sum_{j \in \mathcal{D}} \left( \alpha_j - \sum_{i \in S_j} \beta_{ij} \right) &\leq \sum_{j \in \mathcal{D}} \alpha_j - |\mathcal{IS} \setminus (\mathcal{F}_S^{(\ell)} \cup \mathcal{F}_S^{(\ell+1)})| \left( \lambda - \frac{1}{n} \right) - \sum_{x \in \{\ell, \ell+1\}} \sum_{i \in \mathcal{F}_S^{(x)} \cap \mathcal{IS}} \left( \sum_{j \in \mathcal{D}_S^{(x)}(i)} \beta_{ij}^{(x)} - \frac{1}{n} \right) \\ &\leq \sum_{j \in \mathcal{D}} \alpha_j - |\mathcal{IS} \setminus (\mathcal{F}_S^{(\ell)} \cup \mathcal{F}_S^{(\ell+1)})| \lambda - \sum_{x \in \{\ell, \ell+1\}} \sum_{i \in \mathcal{F}_S^{(x)} \cap \mathcal{IS}} \sum_{j \in \mathcal{D}_S^{(x)}(i)} \beta_{ij}^{(x)} + \frac{|\mathcal{IS}|}{n}. \end{aligned}$$

Since  $\mathcal{S}^{(x)}$  is roundable for  $x \in \{\ell, \ell+1\}$ , we have  $\sum_{i \in \mathcal{F}_S^{(x)}} \sum_{j \in \mathcal{D}_S^{(x)}(i)} \beta_{ij}^{(x)} \geq \lambda |\mathcal{F}_S^{(x)}| - \gamma \cdot \text{OPT}_k$ . Moreover, as  $\alpha^{(x)}$  is a feasible solution of  $\text{DUAL}(\lambda + \frac{1}{n})$ , we have that  $\sum_{j \in \mathcal{D}_S^{(x)}(i)} \beta_{ij}^{(x)} \leq \lambda + \frac{1}{n}$  for any  $i \in \mathcal{F}_S^{(x)}$ . Therefore,

$$\sum_{i \in \mathcal{F}_S^{(x)} \cap \mathcal{IS}} \sum_{j \in \mathcal{D}_S^{(x)}(i)} \beta_{ij}^{(x)} \geq \lambda |\mathcal{F}_S^{(x)} \cap \mathcal{IS}| - \frac{|\mathcal{F}_S^{(x)} \setminus \mathcal{IS}|}{n} - \gamma \cdot \text{OPT}_k \geq \lambda |\mathcal{F}_S^{(x)} \cap \mathcal{IS}| - 2\gamma \cdot \text{OPT}_k.$$

where for the final inequality we use that  $|\mathcal{F}_S^{(x)}| \leq n \leq \text{OPT}_k$ , which follows from Definition 5.1, the fact that any client has distance at least 1 to its closest facility, and  $1/n \ll \gamma$ . Combining this with the above inequalities yields

$$\begin{aligned} \sum_{j \in \mathcal{D}} \left( \alpha_j - \sum_{i \in S_j} \beta_{ij} \right) &\leq \sum_{j \in \mathcal{D}} \alpha_j - |\mathcal{IS}| \lambda + 4\gamma \cdot \text{OPT}_k + \frac{|\mathcal{IS}|}{n} \\ &= \sum_{j \in \mathcal{D}} \alpha_j - |\mathcal{IS}| \left( \lambda + \frac{1}{n} \right) + 4\gamma \cdot \text{OPT}_k + \frac{2|\mathcal{IS}|}{n} \\ &\leq \text{OPT}_k + 4\gamma \cdot \text{OPT}_k + \frac{2|\mathcal{IS}|}{n} \leq (1 + 5\gamma) \text{OPT}_k, \end{aligned}$$

where we in the penultimate inequality used that  $\alpha$  is a feasible solution to  $\text{DUAL}(\lambda + \frac{1}{n})$  and  $|\mathcal{IS}| \geq k$ , therefore  $\sum_{j \in \mathcal{D}} \alpha_j - |\mathcal{IS}| \left( \lambda + \frac{1}{n} \right) \leq \sum_{j \in \mathcal{D}} \alpha_j - k \left( \lambda + \frac{1}{n} \right) \leq \text{OPT}_k$ ; and, in the last inequality, we used that  $\gamma \cdot \text{OPT}_k \geq \gamma n \geq 2$  and the assumption that  $|\mathcal{IS}| \leq n$ .

We conclude the proof by substituting this bound in (6.2):

$$\sum_{j \in \mathcal{D}} d(j, \mathcal{IS})^2 \leq (\rho + 200\epsilon)(1 + 5\gamma) \text{OPT}_k + 36\gamma \cdot \text{OPT}_k \leq (\rho + 1000\epsilon) \text{OPT}_k. \quad \square$$

## 7 The algorithm RAISEPRICE

In this section, we give the details of the algorithm RAISEPRICE, which is responsible for raising facility prices and generating sequences of roundable solutions in Algorithm 1. It is based on similar insights as used in the quasi-polynomial algorithm described in Section 4. Let us first provide a high-level overview of our approach. Recall that in our analysis of that procedure, changing the values  $\alpha_j$  in some bucket  $b$  by  $\epsilon_z$  roughly required changing the values in bucket  $b+1$  by up to  $n\epsilon_z$ . Because there were  $\Omega(\log(n))$  buckets, the total change in the last bucket was potentially  $\epsilon_z n^{\Omega(\log n)}$ , and so to obtain a close sequence of  $\alpha$  values, we required  $\epsilon_z = n^{-\Omega(\log n)}$  in that section. Here,

we reduce the dependence on  $n$  by changing the way in which we increase the opening price  $z$ . As in the quasi-polynomial procedure, our algorithm repeatedly increases the opening cost of every facility from  $\lambda$  to  $\lambda + \epsilon_z$ , for some appropriate small increment  $\epsilon_z = n^{-O(1)} < \epsilon$ . However, instead of performing each such increase for every facility at once, we instead increase only a single facility's price at a time. Each such increase will still cause some clients to become unsatisfied (or undecided as we shall call them), and so we must repair the solution. In contrast to the quasi-polynomial procedure, RAISEPRICE repairs the solution over a series of *stages*. We show this will result in a *polynomial length* sequence of close, roundable solutions.

**Notation:** Throughout this section, we let  $z_i$  denote the current price for a facility  $i \in \mathcal{F}$ , where always  $z_i \in \{\lambda, \lambda + \epsilon_z\}$ . We shall now say that  $i$  is *tight* if  $\sum_{j \in \mathcal{D}} \beta_{ij} = z_i$ , where as before for a solution  $\alpha$ , we use  $\beta_{ij}$  as a shorthand for  $[\alpha_j - d(j, i)^2]^+$ . It will also be convenient to denote  $\sqrt{\alpha_j}$  by  $\bar{\alpha}_j$ . Note that  $\beta_{ij} > 0$ , if and only if  $\bar{\alpha}_j > d(j, i)$ . As in the quasi-polynomial procedure, we shall divide the range of possible values for  $\alpha_j$  into buckets: we define  $B(v) = 1 + \lfloor \log_{1+\epsilon} v \rfloor$  for any  $v \geq 1$  and  $B(v) = 0$ , for all  $v \leq 1$ .

To control the number of undecided (unsatisfied) clients, it will be important to control the way clients may be increased and decreased throughout our algorithm. To accomplish this, we shall not insist that every client has some tight witness in every vector  $\alpha$  that we produce (in contrast to Invariant 1 in the quasi-polynomial algorithm). Rather, we shall consider several different types of clients:

- *witnessed clients*  $j$  have a tight edge to some tight facility  $i$  with  $B(\alpha_j) \geq B(t_i)$ . In this case, we say that  $i$  is a *witness* for  $j$ . Note that if  $i$  is a witness for  $j$  we necessarily have  $(1 + \epsilon)\alpha_j \geq t_i$ .<sup>5</sup>
- *stopped clients*  $j$  have

$$2\bar{\alpha}_j \geq d(j, j') + 6\bar{\alpha}_{j'} \quad (7.1)$$

for some other client  $j'$ . In this case, we say that  $j'$  *stops*  $j$ . Note that if  $j'$  stops  $j$ , we necessarily have  $\bar{\alpha}_j \geq 3\bar{\alpha}_{j'}$  and so  $\alpha_j \geq 9\alpha_{j'}$ .

- *undecided clients*  $j$  are neither witnessed nor stopped.

Let us additionally call any client that is witnessed or stopped *decided*. Note that the sets of witnessed and stopped clients are not necessarily disjoint. However, we have the following lemma, which follows directly from the triangle inequality and our definitions:

**Lemma 7.1.** *Suppose that  $j$  is stopped. Then  $j$  must be stopped by some  $j'$  that is not stopped.*

*Proof.* We proceed by induction over clients  $j$  in non-decreasing order of  $\alpha_j$ . First, note that the client  $j$  with smallest value  $\alpha_j$  cannot be stopped. For the general case, suppose that  $j$  is stopped by some  $j_1$ . Then,  $\alpha_{j_1} < \alpha_j$ . If  $j_1$  is stopped, then by the induction hypothesis it must be stopped by some  $j_2$  that is not stopped. Then, we have  $2\bar{\alpha}_j \geq d(j, j_1) + 6\bar{\alpha}_{j_1}$ , and  $2\bar{\alpha}_{j_1} \geq d(j_1, j_2) + 6\bar{\alpha}_{j_2}$ . It follows that

$$2\bar{\alpha}_j \geq d(j, j_1) + (6 - 2)\bar{\alpha}_{j_1} + d(j_1, j_2) + 6\bar{\alpha}_{j_2} \geq d(j, j_2) + 6\bar{\alpha}_{j_2}.$$

Thus  $j$  is stopped by  $j_2$ , as well. □

---

<sup>5</sup>Here, we use that all  $\alpha$ -values will be at least one and two values in the same bucket differs thus by at most a factor  $1 + \epsilon$ . We also remark that this is the same concept as in Invariant 1 of the quasi-polynomial algorithm.

Intuitively, the stopping criterion will ensure that no  $\alpha_j$  grows too large compared to the  $\alpha$ -values of nearby clients. At the same time it is designed so that all decided clients will have a good approximation guarantee.

Finally, we shall require that the following invariants hold throughout the execution of Algorithm 1.

**Invariant 2** (Feasibility). *For all  $j \in \mathcal{D}$ ,  $\alpha_j \geq 1$  and for all  $i \in \mathcal{F}$ ,  $\sum_{j \in \mathcal{D}} \beta_{ij} \leq z_i$ .*

We remark that for dual feasibility  $\alpha_j \geq 0$  is sufficient but the stronger assumption  $\alpha_j \geq 1$  which is implied by Lemma 4.1 will be convenient.

**Invariant 3** (No strict containment). *For any two clients  $j, j' \in \mathcal{D}$ ,  $\bar{\alpha}_j \leq d(j, j') + \bar{\alpha}_{j'}$ .*

Note that the above invariant says that the ball centered at  $j$  of radius  $\bar{\alpha}_j$  does *not* strictly contain the ball centered at  $j'$  of radius  $\bar{\alpha}_{j'}$ . For future reference, we refer to the ball centered at client  $j$  of radius  $\bar{\alpha}_j$  as the  $\alpha$ -ball of that client.

**Invariant 4** ( $(\alpha^{(0)}, z^{(0)})$  Completely Decided). *Every client is decided in  $(\alpha^{(0)}, z^{(0)})$ .*

Invariant 4 will be maintained as follows (as we show formally in Lemma 8.1): The initial solution satisfies the invariant. Then, given an initial solution  $(\alpha^{(0)}, z^{(0)})$  in which all clients are decided, RAISEPRICE will output a close, roundable sequence  $\mathcal{S}^{(1)}, \dots, \mathcal{S}^{(q)}$ , where  $\mathcal{S}^{(q)} = (\alpha^{(q)}, z^{(q)}, \emptyset, \mathcal{D}_s^{(q)})$  is a roundable solution in which all clients are decided. As the next call to RAISEPRICE will use  $(\alpha^{(q)}, z^{(q)})$  as the initial solution, the invariant is maintained.

## 7.1 The RAISEPRICE procedure.

RAISEPRICE is described in detail in Algorithm 2. Initially, we suppose that we are given a  $\lambda$ -roundable and a completely decided dual solution  $(\alpha^{(0)}, z^{(0)})$  (i.e., satisfying Invariant 4) where  $z_i \in \{\lambda, \lambda + \epsilon_z\}$  for all  $i \in \mathcal{F}$ . Additionally, let  $\text{IS}^{(0)}$  be the independent set  $(\alpha^{(0)}, z^{(0)})$  of the conflict graph  $H^{(0)}$  associated to the roundable solution  $(\alpha^{(0)}, z^{(0)})$ , produced at the end of the previous call to GRAPHUPDATE as described in Algorithm 1. We shall assume that  $|\text{IS}^{(0)}| \geq k$ , as otherwise, Algorithm 1 would have already terminated. For a specified facility  $i^+$ , RAISEPRICE sets  $z_{i^+} \leftarrow z_{i^+} + \epsilon_z$ . This may result in some clients using  $i^+$  as a witness becoming undecided; specifically, those clients that are not stopped and have no witness except  $i^+$  in  $(\alpha^{(0)}, z^{(0)})$ . We let  $U^{(0)}$  to be the set of all these initially undecided clients. Throughout RAISEPRICE, we maintain a set  $U$  of currently undecided clients, and repair the solution over a series of multiple stages, by calling an auxiliary procedure, SWEEP. Each repair stage  $s$  will be associated with a threshold  $\theta_s$ , and will make multiple calls to the procedure SWEEP, each producing a new solution  $\alpha$ . The algorithm RAISEPRICE constructs a roundable solution  $\mathcal{S} = (\alpha, z, \mathcal{F}_s, \mathcal{D}_s)$  from each such  $\alpha$ , and returns the sequence  $\mathcal{S}^{(1)}, \dots, \mathcal{S}^{(q)}$  of all such roundable solutions, in the order they were constructed. RAISEPRICE terminates once it constructs some solution in which all clients are decided. In Section 8, we shall show that this must happen after at most  $O(\log n)$  stages, and that each stage requires only a polynomial number of calls to SWEEP. In addition, we show that the produced sequence is close and roundable.

Before describing SWEEP in detail, let us first provide some intuition for the selection of the thresholds  $\theta_s$  and describe the construction of each roundable solution  $\mathcal{S}^{(\ell)} = (\alpha^{(\ell)}, z^{(\ell)}, \mathcal{F}_s^{(\ell)}, \mathcal{D}_s^{(\ell)})$  in RAISEPRICE. Our procedure SWEEP will adjust client values  $\alpha_j$  similarly to the procedure QUASISWEEP described in Section 4.1. However, in each stage  $s$ , we ensure that SWEEP never

---

**Algorithm 2:** RAISEPRICE( $\alpha^{(0)}, z^{(0)}, \text{IS}^{(0)}, i^+$ )

---

**Input:**  $(\alpha^{(0)}, z^{(0)})$  : a  $\lambda$ -roundable solution satisfying Invariant 4 and each  $z_i \in \{\lambda, \lambda + \epsilon_z\}$ .  
 $\text{IS}^{(0)}$  : the independent set of conflict graph  $H^{(0)}$  produced by GRAPHUPDATE.

$i^+$  : a facility whose price  $z_{i^+}$  is being increased from  $\lambda$  to  $\lambda + \epsilon_z$

**Output:** Sequence  $\mathcal{S}^{(1)} = (\alpha^{(1)}, z^{(1)}, \mathcal{F}_s^{(1)}, \mathcal{D}_s^{(1)}), \dots, \mathcal{S}^{(q)} = (\alpha^{(q)}, z^{(q)}, \mathcal{F}_s^{(q)}, \mathcal{D}_s^{(q)})$  of close  $\lambda$ -roundable solutions, where all clients are decided in  $\mathcal{S}^{(q)}$ .

```

1   $(\alpha, z) \leftarrow (\alpha^{(0)}, z^{(0)})$ 
2   $z_{i^+} \leftarrow z_{i^+} + \epsilon_z$ 
3  Let  $U^{(0)}$  be the set of clients now undecided.
4  Set  $K = \Theta(\epsilon^{-1}\gamma^{-4})$  and select a shift parameter  $0 \leq \sigma < K/2$ .
5  Set  $\theta_1 = (\max_{j \in U^{(0)}} \alpha_j^{(0)} + 2\epsilon_z)(1 + \epsilon)^\sigma$  and  $\theta_s = (1 + \epsilon)^K \theta_{s-1}$  for all  $s > 1$ .
6   $U \leftarrow U^{(0)}$ 
7   $\ell \leftarrow 1, s \leftarrow 1$ 
8  while  $U \neq \emptyset$  do
    /* Execute repair stage  $s$  */
9    while there is some  $j \in U$  with  $\alpha_j < \theta_s$  do
10      $\alpha \leftarrow \text{SWEEP}(\theta_s, \alpha)$  (this procedure is described in Section 7.2)
11      $U \leftarrow$  set of clients now undecided.
12     Form  $\mathcal{F}_s$  and  $\mathcal{D}_s$  using  $\alpha, z, \alpha^{(0)}$ , and  $\text{IS}^{(0)}$ .
13      $\mathcal{S}^{(\ell)} \leftarrow (\alpha, z, \mathcal{F}_s, \mathcal{D}_s)$ .
14      $\ell \leftarrow \ell + 1$ 
15    $s \leftarrow s + 1$ 

```

---

increases any  $\alpha_j$  above the threshold  $\theta_s$  beyond its initial value  $\alpha_j^{(0)}$ , i.e., we ensure that  $\alpha_j \leq \alpha_j^{(0)}$  for any  $\alpha_j \geq \theta_s$ . We set

$$\theta_1 = (\max_{j \in U^{(0)}} \alpha_j^{(0)} + 2\epsilon_z)(1 + \epsilon)^\sigma \quad \text{and} \quad \theta_s = (1 + \epsilon)^K \theta_{s-1},$$

where  $K = \Theta(\epsilon^{-1}\gamma^{-4})$  is an integer parameter and  $\sigma$  is a integer “shift” parameter chosen uniformly at random<sup>6</sup> from  $[0, K/2)$ . Our selection of thresholds ensures that each stage updates only those  $\alpha_j$  in a constant  $K$  number of buckets. Thus, the total change in any  $\alpha$ -value will be at most  $n^{O(K)}$ , which will allow us to obtain a polynomial running time. This comes at the price of some clients remaining undecided after each stage, and some such clients  $j$  may have service cost much higher than  $\rho \cdot \alpha_j$ . We let  $\mathcal{B}$  denote the set of all such “bad” clients. Using that the  $\alpha$ -values are relatively well-behaved throughout RAISEPRICE, we show that only those clients  $j$  with  $\alpha_j^{(0)}$  relatively near to the threshold  $\theta_s$  can be added to  $\mathcal{B}$  in stage  $s$ . Then, the random shift  $\sigma$  in choosing our definition of thresholds will allow us to show that only an  $O(K^{-1})$  fraction of clients can be bad throughout RAISEPRICE. Moreover, we can bound the cost of each client  $j \in \mathcal{B}$  by  $36\alpha_j^{(0)}$ . Intuitively, then, if at least a constant fraction of each  $\alpha_j^{(0)}$  is contributing to the service cost  $c(j, \text{IS}^{(0)})$ , then we can bound the effect of these bad clients by setting  $K$  to be a sufficiently large constant, then using Theorem 6.4 to conclude that:

$$\sum_{j \in \mathcal{B}} 36\alpha_j^{(0)} \leq \epsilon \cdot \sum_{j \in \mathcal{D}} c(j, \text{IS}^{(0)}) \leq O(\epsilon) \cdot \text{OPT}_k.$$

---

<sup>6</sup>We shall show that it is in fact easy to select an appropriate  $\sigma$  deterministically (see Remark 8.16).

Unfortunately, it may happen that many clients  $j \in \mathcal{B}$  have  $\alpha_j^{(0)} - c(j, \mathbf{IS}^{(0)}) \approx \alpha_j^{(0)}$ . That is, some clients may be using almost all of their  $\alpha^{(0)}$ -values to pay for the opening costs of facilities. In this case, we could have  $\sum_{j \in \mathcal{B}} \alpha_j^{(0)}$  arbitrarily larger than  $\sum_{j \in \mathcal{D}} (j, \mathbf{IS}^{(0)})$ . In order to cope with this situation, we introduce a notion of *dense* clients and facilities in Section 8.5. These troublesome clients and facilities are handled by carefully constructing the remaining components  $\mathcal{F}_s$  and  $\mathcal{D}_s$  of the roundable solution in line 12. We defer the formal details to Section 8.5, but the intuition is if enough bad clients are paying mostly for the opening cost of a facility, then we can afford to open this facility *even if it is not tight*. This is precisely the role of special facilities in Definition 5.1.

## 7.2 The SWEEP Procedure

It remains to describe our last procedure, SWEEP in more detail. SWEEP operates in some stage  $s$ , with corresponding threshold value  $\theta_s$ , takes as input the previous  $\alpha$  produced by the algorithm, and produces a new  $\alpha$ . Note that in every call to SWEEP, we let  $\alpha^{(0)}$  denote the roundable solution passed to RAISEPRICE, and  $U$  is the set of undecided clients immediately before SWEEP was called. Just like QUASISWEEP, the procedure SWEEP, maintains a current set of active clients  $A$  and a current threshold  $\theta$ , where initially,  $A = \emptyset$ , and  $\theta = 0$ . We slowly increase  $\theta$  and whenever  $\theta = \alpha_j$  for some client  $j$ , we add  $j$  to  $A$ . While  $j \in A$ , we increase  $\alpha_j$  with  $\theta$ . However, in contrast to QUASISWEEP, SWEEP removes a client  $j$  from  $A$ , whenever one of the following five events occurs:

Rule 1.  $j$  has some witness  $i$ .

Rule 2.  $j$  is stopped by some client  $j'$ .

Rule 3.  $j \in U$  and  $\alpha_j$  is  $\epsilon_z$  larger than its value at the start of SWEEP.

Rule 4.  $\alpha_j \geq \theta_s$  and  $\alpha_j \geq \alpha_j^{(0)}$ .

Rule 5. There is a client  $j'$  that has already been removed from  $A$  such that  $\bar{\alpha}_j \geq d(j, j') + \bar{\alpha}_{j'}$ .

We remark that Rule 5 says that  $j$  is removed from  $A$  as soon as its  $\alpha$ -ball contains the  $\alpha$ -ball of another client  $j'$  that is not currently in  $A$ . This rule is designed so that the algorithm maintains Invariant 3. Also note that if a client  $j$  satisfies one of these conditions when it is added to  $A$ , then we remove  $j$  from  $A$  immediately after it is added. In this case,  $\alpha_j$  is not increased.

As in QUASISWEEP, increasing the values  $\alpha_j$  for clients in  $A$  may cause  $\sum_{j \in \mathcal{D}} \beta_{ij}$  to exceed  $z_i$  for some facility  $i$ . We again handle this by decreasing some other values  $\alpha_{j'}$ . However, here we are more careful in our choice of clients to decrease. Let us call a facility  $i$  *potentially tight* if one of the following conditions hold:

- There is some  $j \in N(i)$  with  $\alpha_j > \alpha_j^{(0)}$ .
- For all  $j \in N^{(0)}(i)$ ,  $\alpha_j \geq \alpha_j^{(0)}$ .

We now decrease  $\alpha_{j'}$  if and only if  $B(\alpha_{j'}) > B(\theta)$  and additionally: for some potentially tight facility  $i$  with  $|N(i) \cap A| \geq 1$ , we have  $\alpha_{j'} = t_i$ . We decrease each such  $\alpha_{j'}$  at a rate of  $|A|$  times the rate that  $\theta$  is increasing. To see that this maintains feasibility we observe that at any time there are  $|A \cap N(i)|$  clients whose contribution to facility  $i$  is increasing, and these contributions are increasing at the same rate as  $\theta$ . Suppose that  $i$  is tight at some moment with some  $j \in N(i) \cap A$ . Then, since  $z_i \geq z_i^{(0)}$ , there must be either at least one client  $j'$  with  $\alpha_{j'} > \alpha_{j'}^{(0)}$  or  $\alpha_{j'} = \alpha_{j'}^{(0)}$  for all  $j' \in N^{(0)}(i)$ , and so  $i$  must be potentially tight. Consider some client  $j_0$  with  $\alpha_{j_0} = t_i$  and note



that  $B(\alpha_{j_0}) = B(t_i) > B(\alpha_j) = B(\theta)$ , since otherwise we would remove  $j$  from  $A$  by Rule 1. The value of  $\alpha_{j_0}$  is currently decreasing at a rate of  $|A| \geq |N(i) \cap A|$  times the rate that  $\theta$  is increasing. Thus, the total contribution to any tight facility  $i$  is never increased.

As in QUASISWEEP, we stop increasing  $\theta$  once every client  $j$  has been added and removed from  $A$ , and then output the resulting  $\alpha$ . Note that SWEEP never changes any  $\alpha_j < \theta$ . In particular, once some  $j$  has been removed from  $A$  it is not subsequently changed. Additionally, observe that once  $B(\theta) \geq B(\alpha_j)$ , SWEEP will not decrease  $\alpha_j$ .

## 8 Analysis of the polynomial-time algorithm

Unfortunately, in contrast to the quasi-polynomial time procedure, here our analysis is quite involved. Let us first provide a high-level overview of our overall approach. Note that any solution that does not contain any undecided clients must necessarily be roundable with  $\mathcal{F}_S = \emptyset$ , and  $\mathcal{D}_B = \emptyset$ . Indeed, for any witnessed client  $j$  there is a tight facility  $i$  with  $(1 + \epsilon)\bar{\alpha}_j \geq \sqrt{t_i}$  and  $\bar{\alpha}_j \geq d(j, i)$  and so

$$(1 + (1 + \epsilon)\sqrt{\delta})\bar{\alpha}_j \geq d(j, i) + \sqrt{\delta t_i}.$$

Similarly, any stopped client  $j$  in such a solution must be stopped by some witnessed  $j'$  (using Lemma 7.1). Let  $i$  be the witness of  $j'$ . Then,

$$d(j, i) + \sqrt{\delta t_i} \leq d(j, j') + d(j', i) + \sqrt{\delta t_i} \leq 2\bar{\alpha}_j - 6\bar{\alpha}_{j'} + (1 + (1 + \epsilon)\sqrt{\delta})\bar{\alpha}_{j'} < (1 + \sqrt{\delta})\bar{\alpha}_j,$$

since  $\sqrt{\delta} \geq 1$ . As  $\tau_i \leq t_i$  for any facility  $i \in \mathcal{F}$ , the required inequalities from Definition 5.1 hold for any decided client  $j$ . In the following our main goal will then be to bound the cost of the remaining, undecided clients in any solution produced by RAISEPRICE.

Our first task is to characterize which clients may *currently* be undecided. To this end, we first prove some basic properties about the way SWEEP alters  $\alpha$ -values together with Invariants 2, 3, and 4 (Section 8.1). Then, we show that only clients above threshold  $\theta_s$  in each stage  $s$  can become undecided (Section 8.2). In Section 8.3, we bound the cost of all decided and undecided clients, showing that we can indeed obtain a  $(\rho + O(\epsilon))$ -approximation for all decided clients and a slightly worse guarantee for undecided ones. Next, we would like to argue that most clients have good connection cost. Specifically, we would like to choose a set of thresholds that ensure that only a constant fraction of clients become undecided throughout the *entirety* of RAISEPRICE. In order to accomplish this, we show that our  $\alpha$ -values remain relatively stable throughout RAISEPRICE (Section 8.4). This allows us to prove that RAISEPRICE outputs close solutions and to characterize those clients that may become undecided in RAISEPRICE by their values  $\alpha_j^{(0)}$  at the *beginning* of RAISEPRICE. This, together with our selection of thresholds, ensures that only an arbitrarily small, constant fraction of clients do not have the desired guarantee. However, we must also show that these clients do not contribute more than a constant to  $\text{OPT}_k$ . As discussed above, this will follow immediately from our analysis for those clients whose service cost is at least a constant fraction of  $\alpha_j^{(0)}$ . For other (i.e. *dense*) clients, we must use a different argument, involving the sets of special facilities and clients  $\mathcal{F}_s$  and  $\mathcal{D}_s(i)$  (Section 8.5). Finally, we put all of these pieces together and show that RAISEPRICE produces a close sequence of polynomially many roundable solutions and runs in polynomial time (Section 8.6).

### 8.1 Basic properties of SWEEP and Invariants 2, 3, and 4

We start by showing that Invariants 2, 3, and 4 hold.

**Lemma 8.1.** *Invariants 2, 3, and 4 hold throughout Algorithm 1.*

*Proof.* We begin by proving Invariant 2, i.e., that the algorithm maintains a feasible dual solution  $\alpha$  with the additional property that  $\alpha_j \geq 1$  for all  $j \in \mathcal{D}$ . Recall our construction of the initial solution  $\alpha^{(0)}$  for Algorithm 1: we set  $\alpha_j = 0$  for all  $j \in \mathcal{D}$  and then increase all  $\alpha_j$  at a uniform rate. We stop increasing a value  $\alpha_j$  whenever  $j$  gains a tight edge to some facility  $i \in \mathcal{F}$  or  $2\bar{\alpha}_j \geq d(j, j') + 6\bar{\alpha}_{j'}$  for some  $j' \in \mathcal{D}$ . Note that no  $\alpha_j$  is increased after  $\alpha_j = d(j, i)^2$  for some facility  $i$ . Thus, we have  $\beta_{ij}^{(0)} = 0$  for all  $j \in \mathcal{D}$  and  $i \in \mathcal{F}$ , and so  $\alpha^{(0)}$  is feasible. Now, we show that  $\min_{j \in \mathcal{D}} \alpha_j^{(0)} \geq 1$ . Consider the client  $j_0$  that first stops increasing in our greedy initialization process. At the time  $\alpha_{j_0}$  stops increasing, we have  $\alpha_j = \alpha_{j_0}$  for all  $j \in \mathcal{D}$  and so  $2\bar{\alpha}_j \geq d(j, j') + 6\bar{\alpha}_{j'}$  cannot hold for any pair  $j, j'$  of clients. Thus,  $j_0$  must have stopped increasing because  $\alpha_{j_0} = d(j_0, i)^2$  for some facility  $i$ . By our preprocessing (Lemma 4.1) we have  $d(j_0, i)^2 \geq 1$ , and so  $\alpha_{j_0}^{(0)} \geq 1$ . Moreover,  $\alpha_{j_0}^{(0)} = \min_{j \in \mathcal{D}} \alpha_j^{(0)}$ , and so indeed  $\alpha_j^{(0)} \geq 1$  for all  $j \in \mathcal{D}$ . Now, we show that Algorithm 1 preserves Invariant 2. Note that  $\alpha$  is altered only by subroutine SWEEP, and by construction, SWEEP ensures that always  $\sum_j \beta_{ij} \leq z_i$ . Moreover, SWEEP decreases any  $\alpha_j$  only while it is increasing some other  $\alpha_{j'} < \alpha_j$  such that  $j'$  has a tight edge to some facility  $i$ . By our preprocessing (Lemma 4.1)  $\alpha_{j'} \geq d(j', i)^2 \geq 1$  for any such  $j'$ . Thus, no  $\alpha_j$  is ever decreased below 1.

Next, we prove Invariant 3, i.e., that no client's  $\alpha$ -ball is strictly contained in the  $\alpha$ -ball of another client. First, let us show that the initially constructed solution  $(\alpha^{(0)}, z^{(0)})$  satisfies Invariant 3. Note that  $\alpha_j^{(0)}$  is equal to the value of  $\alpha_j$  at the time that our initialization procedure stopped increasing  $\alpha_j$ . Consider any pair of clients  $j$  and  $j'$ . If  $\alpha_j^{(0)} \leq \alpha_{j'}^{(0)}$  then clearly  $\bar{\alpha}_j^{(0)} \leq d(j, j') + \bar{\alpha}_{j'}^{(0)}$ . Thus, suppose that  $\alpha_j^{(0)} > \alpha_{j'}^{(0)}$ , so  $\alpha_{j'}$  stopped increasing before  $\alpha_j$  in our initialization procedure. If  $\alpha_{j'}$  stopped increasing because  $j'$  gained a tight edge to a facility  $i$ , then once  $\bar{\alpha}_j = d(j, j') + \bar{\alpha}_{j'}$ ,  $j$  will have a tight edge to  $i$  and stop increasing. If  $\alpha_{j'}$  stopped increasing because  $2\bar{\alpha}_{j'} = d(j', j'') + 6\bar{\alpha}_{j''}$  for some client  $j''$ , then when  $\bar{\alpha}_j = d(j, j') + \bar{\alpha}_{j'}$  we will have

$$2\bar{\alpha}_j = 2d(j, j') + 2\bar{\alpha}_{j'} = 2d(j, j') + d(j', j'') + 6\bar{\alpha}_{j''} \geq d(j, j'') + 6\bar{\alpha}_{j''}$$

and so  $\alpha_j$  must stop increasing. In any case, we must have  $\bar{\alpha}_j^{(0)} \leq d(j, j') + \bar{\alpha}_{j'}^{(0)}$ . Having shown that the invariant is true for the first  $\alpha^{(0)}$  constructed in Algorithm 1, let us now prove that it is maintained. First, we show that the inequality  $\bar{\alpha}_j \leq d(j, j') + \bar{\alpha}_{j'}$  will not be violated by increasing  $\bar{\alpha}_j$ . Suppose that  $j \in A$  and so  $\alpha_j$  is increasing. As long as  $j' \in A$ , as well, we have  $\alpha_j = \alpha_{j'} = \theta$ , and so  $\bar{\alpha}_j \leq d(j, j') + \bar{\alpha}_{j'}$ . On the other hand, if  $j' \notin A$ , then as soon as  $\bar{\alpha}_j = d(j, j') + \bar{\alpha}_{j'}$ ,  $j$  will be removed from  $A$  by Rule 5 and  $\bar{\alpha}_j$  will no longer increase. Now we show that also  $\bar{\alpha}_j \leq d(j, j') + \bar{\alpha}_{j'}$  will not be violated by decreasing  $\alpha_{j'}$ . Suppose that  $\alpha_{j'}$  is decreasing. Then, there must be some potentially tight facility  $i$  with  $j' \in N(i)$  and  $t_i = \alpha_{j'}$ . Let  $i$  be any such facility. If at some point we have  $\bar{\alpha}_j = d(j, j') + \bar{\alpha}_{j'}$ , then we must also have  $j \in N(i)$  at this moment and  $\alpha_j \geq \alpha_{j'} = t_i$ . Thus,  $\alpha_j$  is also decreasing and in fact  $\alpha_j = \alpha_{j'}$  (since also  $\alpha_j \leq t_i$ ). Then,  $\bar{\alpha}_j$  and  $\bar{\alpha}_{j'}$  are decreasing at same rate and so  $\bar{\alpha}_j = d(j, j') + \bar{\alpha}_{j'}$  as long as  $\bar{\alpha}_{j'}$  continues to decrease.

Finally, we prove Invariant 4, i.e., that the input solution  $(\alpha^{(0)}, z^{(0)})$  to RAISEPRICE is always completely decided. Every client  $j$  is either stopped by some client  $j'$  or has a tight edge to some facility  $i$  in our initially constructed solution  $(\alpha^{(0)}, z^{(0)})$ . Moreover, the initialization process ensures that  $N(i) = \emptyset$  for all  $i$  (since  $\beta_{ij} = 0$  for all  $i \in \mathcal{F}$  and  $j \in \mathcal{D}$ ). Thus, in the latter case  $t_i = 0$  and so  $i$  is in fact a witness for  $j$ , and so every client  $j$  is indeed either stopped or witnessed in this initial solution  $(\alpha^{(0)}, z^{(0)})$ . To show that Invariant 4 holds throughout the rest of the Algorithm 1, we note that  $(\alpha^{(0)}, z^{(0)})$  is always updated (in line 9 of Algorithm 1 where  $\mathcal{S}^{(0)} \leftarrow \mathcal{S}^{(q)}$ ) with the  $\alpha$ -values corresponding to the last solution produced in a call to RAISEPRICE. Due to the condition in the main loop of RAISEPRICE, every client is decided in this solution.  $\square$

The next lemma makes some basic observations about the way in which SWEEP alters the  $\alpha$ -values.

**Lemma 8.2.** *The procedure SWEEP satisfies the following properties:*

- Property 1. *Any client  $j$  that becomes decided after being added to  $A$  remains decided until the end of SWEEP.*
- Property 2. *If the  $\alpha$ -ball of a client  $j$  contains the  $\alpha$ -ball of a decided client, then  $j$  is decided.*
- Property 3. *Consider the solution  $\alpha$  at the beginning of SWEEP, and let  $\mu = \min_{j' \in U} \alpha_{j'}$ . Then, no  $\alpha_j < \mu$  is increased by SWEEP, and no  $\alpha_j$  with  $B(\alpha_j) \leq B(\mu)$  is decreased by SWEEP.*

*Proof.* For Property 1, suppose first that  $j$  had a witness  $i$  at some point after being added to  $A$ . Consider any  $j' \in N(i)$  at this moment. At this moment, we must have  $B(\alpha_{j'}) \leq B(\alpha_j) \leq B(\theta)$  and so  $\alpha_{j'}$  cannot be decreased for the remainder of SWEEP. In particular,  $j$  retains a tight edge to  $i$  until the end of SWEEP and  $i$  remains tight until the end of SWEEP. Additionally, any client  $j'$  with  $\alpha_{j'} > t_i$  will be removed from  $A$  as soon as it gains a tight edge to  $i$  (by Rule 1 since  $i$  would then be a witness for  $j'$ ). Thus,  $t_i$  cannot increase and so  $i$  remains a witness for  $j$  until the end of SWEEP. Next, suppose that  $j$  was stopped by some  $j'$  after being added to  $A$ . Then, at this moment,  $\alpha_{j'} < \alpha_j \leq \theta$ . Hence, for the remainder of SWEEP, neither  $\alpha_{j'}$  or  $\alpha_j$  are changed and so  $j$  remains stopped by  $j'$ .

For Property 2 suppose that the  $\alpha$ -ball of client  $j$  contains the  $\alpha$ -ball of a decided client  $j'$ . Then if  $j'$  has a witness  $i$ , then  $i$  is also a witness for  $j$ , since  $\bar{\alpha}_j \geq d(j, j') + \bar{\alpha}_{j'} \geq d(j, j') + d(j', i) \geq d(j, i)$  and  $B(\alpha_j) \geq B(\alpha_{j'}) \geq B(t_i)$ . Similarly if  $j'$  is stopped by some client  $j''$  then

$$2\bar{\alpha}_j \geq 2(d(j, j') + \bar{\alpha}_{j'}) \geq 2d(j, j') + 6\bar{\alpha}_{j''} + d(j', j'') \geq 6\bar{\alpha}_{j''} + d(j, j''),$$

and so  $j$  is also stopped by  $j''$ .

Finally, for Property 3, consider the first client  $j$  whose value  $\alpha_j$  is increased by SWEEP. Note that  $j$  must not be decided before calling SWEEP: otherwise, since no other  $\alpha$ -value has yet been changed, this would hold at the moment  $j$  was added to  $A$ , as well, and so  $j$  would immediately be removed by Rule 1 or 2. Thus, the first  $\alpha_j$  that is increased by SWEEP must correspond to some  $j \in U$ , and at the moment this occurs,  $\theta = \alpha_j \geq \mu$ . Furthermore, by the definition of SWEEP, no  $\alpha_j$  can then be decreased unless  $B(\alpha_j) \geq B(\mu) + 1$ .  $\square$

## 8.2 Characterizing currently undecided clients

The next observations follow rather directly from the properties given in Lemma 8.2 and the invariants. These facts will help us bound the number of clients that can become bad throughout the algorithm, and also the total number of calls to SWEEP that must be executed in each call to RAISEPRICE. Throughout this section, we consider a single call to RAISEPRICE and let  $(\alpha^{(0)}, z^{(0)}, \text{IS}^{(0)}, i^+)$  be its input.

**Lemma 8.3.** *In stage 1, SWEEP is executed only a single time. After this call, for every  $j \in U^{(0)}$ , we have  $\alpha_j \leq \alpha_j^{(0)} + \epsilon_z < \theta_1$  and  $j$  is decided.*

*Proof.* Consider any client  $j_0 \in U^{(0)}$ . Then  $i^+$  was  $j_0$ 's witness in  $(\alpha^{(0)}, z^{(0)})$ , and  $j_0$  must not have been stopped or have had any other witness  $i \neq i^+$ . Observe that our choice of  $\theta_1$  ensures that

$\alpha_{j_0}^{(0)} + \epsilon_z < \theta_1$ , so any  $j_0 \in U^{(0)}$  will be removed from  $A$  by Rule 3 once  $\alpha_j = \alpha_j^{(0)} + \epsilon_z$ . Thus we must  $\alpha_j \leq \alpha_{j_0}^{(0)} + \epsilon_z < \theta_1$  at the end of SWEEP for every  $j_0 \in U^{(0)}$ .

This also implies that no such  $j_0$  is removed from  $A$  by Rule 4. We now show that when  $j_0$  is removed from  $A$  by any other rule, it must be decided. By Property 1,  $j_0$  is then decided at the end of SWEEP, as well. First, we observe that if  $j_0$  is removed from  $A$  by Rules 1 or 2, then it is decided by definition. Next, suppose that  $j_0$  was removed by Rule 3, and let  $\mu = \min_{j \in U^{(0)}} \alpha_j^{(0)}$ . Since  $i^+$  was a witness for every  $j \in U^{(0)}$ , we must have  $B(\alpha_j^{(0)}) \leq B(\mu)$  for all  $j \in N^{(0)}(i^+)$ . Thus, by Property 3 of SWEEP,  $\alpha_j \geq \alpha_j^{(0)}$  for every  $j \in N^{(0)}(i^+)$ . Then, since  $\alpha_{j_0} = \alpha_{j_0}^{(0)} + \epsilon_z$ , at the time  $j_0$  was removed from  $A$ ,  $i^+$  must have been tight and also a witness for  $j_0$ . By Property 1,  $j_0$  then remains decided until the end of SWEEP. Finally, we consider the case in which  $j_0$  was removed by Rule 5. We show the following:

**Claim.** *Suppose that some client  $j$  is removed from  $A$  by Rule 5 and that  $j$  is undecided at this time. Then,  $\alpha_j \geq \theta_1$ .*

*Proof.* Consider the first time that any client  $j$  that is undecided is removed from  $A$  by Rule 5. By Property 2, the  $\alpha$ -ball of this client  $j$  must contain the  $\alpha$ -ball of some undecided client  $j'$  that was previously removed from  $A$ . By Property 1 and our choice of time,  $j'$  must have been removed from  $A$  by Rule 3 or 4. However, if  $j'$  was removed by Rule 3, we must have  $j' \in U^{(0)}$  and so, as we have previously shown,  $j'$  must be decided. Thus,  $j'$  was removed by Rule 4, and so presently  $\alpha_j = \theta \geq \alpha_{j'} \geq \theta_1$ . To complete the proof, we observe that any client that is removed from  $A$  after  $j$  must have an  $\alpha$ -value at least  $\alpha_j \geq \theta_1$ .  $\square$

It follows by the above Claim that no  $j_0 \in U^{(0)}$  can be undecided when it is removed by Rule 5, since, as we have shown,  $\alpha_{j_0} < \theta_1$  for all such  $j_0$ . By the above cases, every client  $j_0 \in U^{(0)}$  is decided with  $\alpha_j \leq \alpha_{j_0}^{(0)} + \epsilon_z < \theta_1$  at the end of SWEEP.

It remains to show that RAISEPRICE continues to stage 2 after one call to SWEEP. Consider some client  $j$  that is undecided at the end of SWEEP. By Property 1  $j$  must not have been removed from  $A$  by Rule 1 or Rule 2. Moreover, we must have  $j \notin U^{(0)}$  and so  $j$  was not removed from  $A$  by Rule 3. Thus,  $j$  was removed from  $A$  by Rule 4 or 5. In either case,  $\alpha_j \geq \theta_1$  at this moment (and so also at the end of SWEEP, since no  $\alpha_j$  is changed after  $j$  is removed from  $A$ ). Thus, after the first call to SWEEP in stage 1, every undecided client  $j$  has  $\alpha_j \geq \theta_1$  and so RAISEPRICE immediately continues to stage 2.  $\square$

**Lemma 8.4.** *Consider any solution  $(\alpha, z)$  produced by RAISEPRICE. If  $j$  is undecided in  $(\alpha, z)$ , then  $\alpha_j \geq \alpha_j^{(0)}$ .*

*Proof.* Suppose toward contradiction that the statement is false. Consider the first call to SWEEP that produces a solution violating it and for this call let  $j$  be the first client (in the order of removal from  $A$ ) such that  $\alpha_j < \alpha_j^{(0)}$  when  $j$  is removed from  $A$  but  $j$  is undecided<sup>7</sup>. Then since,  $j$  is undecided it was removed by Rule 3, 4, or 5. If  $j$  was removed by Rule 4, then at this moment  $\alpha_j \geq \alpha_j^{(0)}$ . Suppose then that  $j$  was removed by Rule 3. Then,  $j \in U$ . By Lemma 8.3, no client  $j \in U$  before the first call to SWEEP is undecided after this call, so  $j$  must have been undecided at the end of some preceding call to SWEEP. By assumption, we must have had  $\alpha_j \geq \alpha_j^{(0)}$  at the moment  $j$  was removed from  $A$  in this preceding call (and so also immediately before the present

---

<sup>7</sup>By Property 1, any client  $j$  violating the statement must be undecided when removed from  $A$  and have  $\alpha_j < \alpha_j^{(0)}$  at the time of its removal from  $A$  since SWEEP does not change  $j$ 's  $\alpha$ -value thereafter.

call). But,  $\alpha_j$  has increased by  $\epsilon_z$ , so still  $\alpha_j \geq \alpha_j^{(0)}$ . Finally, suppose  $j$  was removed by Rule 5. Then, the  $\alpha$ -ball of  $j$  must contain the  $\alpha$ -ball of a client  $j'$  that has already been removed  $A$ . If  $j'$  is decided, then by Property 2  $j$  is decided as well. Suppose that  $j'$  is undecided. Then, since we picked the first client that violated the condition of the lemma, and  $j'$  was already removed from  $A$ , we have that  $\alpha_{j'} \geq \alpha_{j'}^{(0)}$ . But then, if  $\alpha_j < \alpha_j^{(0)}$ , we have  $\bar{\alpha}_j^{(0)} > \bar{\alpha}_j \geq d(j, j') + \bar{\alpha}_{j'} \geq d(j, j') + \bar{\alpha}_{j'}^{(0)}$ , which contradicts Invariant 3. In all cases we showed that we must have  $\alpha_j \geq \alpha_j^{(0)}$  at the moment that  $j$  was removed from  $A$ , and so also at the end of SWEEP, contradicting our assumption that  $\alpha_j < \alpha_j^{(0)}$  for some undecided client  $j$ .  $\square$

**Lemma 8.5.** *In every stage  $s > 1$ , no  $\alpha_j$  is changed by SWEEP until  $\theta \geq \theta_{s-1}$ . In particular, every client  $j$  with  $\alpha_j < \theta_{s-1}$  is decided for every solution produced by RAISEPRICE in stage  $s > 1$ .*

*Proof.* By Property 3 of SWEEP, no  $\alpha_j$  is changed until  $\theta = \min_{j \in U} \alpha_j$ . Thus to prove the first part of the claim, it suffices to show that in every stage  $s > 1$ , if  $U \neq \emptyset$  then  $\min_{j \in U} \alpha_j \geq \theta_{s-1}$ . Note that the second part of the claim then follows as well for every solution except the one produced by the final call to SWEEP in RAISEPRICE, and this last solution has no undecided clients by Invariant 4.

Let us now prove that  $\min_{j \in U} \alpha_j \geq \theta_{s-1}$  in every stage  $s > 1$ . We proceed by induction on the number of calls to SWEEP made in stage  $s$ . Before the first call to SWEEP in stage  $s$ , we must have  $\alpha_j \geq \theta_{s-1}$  for every  $j \in U$ , since otherwise stage  $s - 1$  would have continued. So, consider some later call to SWEEP in stage  $s$ , and consider any  $j \in U$  before this call. Then, we must have had  $j$  undecided after the preceding call to SWEEP in stage  $s$ . Moreover, by Property 1,  $j$  must have been undecided when it was removed from  $A$  in this preceding call. Consider the *first* client  $j$  that was undecided upon removal from  $A$  in this preceding call. Then,  $j$  cannot have been removed by Rules 1 or 2. Moreover, since every client that has been removed from  $A$  before  $j$  is decided, Property 2 implies that  $j$  must not have been removed by Rule 5. If  $j$  was removed by Rule 3, then we must have had  $j \in U$  already in this preceding call to SWEEP, and so by the induction hypothesis,  $\alpha_j \geq \theta_{s-1}$ . Then, since  $j$  was removed from  $A$  by Rule 3, we had  $\alpha_j \geq \theta_{s-1} + \epsilon_z$ . Finally, if  $j$  was removed by Rule 4, then we must have  $\alpha_j \geq \theta_s > \theta_{s-1}$  by definition. Thus, throughout every stage  $s > 1$ , if  $U \neq \emptyset$ , then  $\min_{j \in U} \alpha_j \geq \theta_{s-1}$ , as desired.  $\square$

**Corollary 8.6.** *Suppose that in  $(\alpha^{(0)}, z^{(0)})$ ,  $j$  is not stopped and has only  $i^+$  as a witness, i.e.,  $j \in U^{(0)}$ . Then, we have that  $j$  is decided with  $\alpha_j \leq \alpha_j^{(0)} + \epsilon_z$  in every solution  $(\alpha, z)$  produced by RAISEPRICE( $\alpha^{(0)}, z^{(0)}, \text{IS}^{(0)}, i^+$ ).*

*Proof.* We have  $j \in U^{(0)}$  and so by Lemma 8.3,  $j$  is decided with  $\alpha_j \leq \alpha_j^{(0)} + \epsilon_z < \theta_1$  in the first solution produced by RAISEPRICE. Moreover, by Lemma 8.5,  $\alpha_j$  remains unchanged and  $j$  remains decided in all later stages.  $\square$

### 8.3 Bounding the cost of clients

In this section we derive inequalities that are used to bound the service cost of each  $(\alpha, z)$  produced during the algorithm. Consider some solution  $\alpha$  produced by the algorithm, and define

$$\mathcal{B} = \{j \in \mathcal{D} : j \text{ is undecided and } 2\bar{\alpha}_j < d(j, j') + 6\bar{\alpha}_{j'}^{(0)} \text{ for all clients } j'\}. \quad (8.1)$$

The set  $\mathcal{B}$  is defined to contain those clients that are (potentially) bad, i.e., have worse connection cost than our target guarantee. Specifically, we now show that all clients  $j \in \mathcal{D} \setminus \mathcal{B}$ , satisfy the first inequality of Property 2 in Definition 5.1 (with  $\tau_i$  replaced by  $t_i$ ), while all clients (in particular those in  $\mathcal{B}$ ) satisfy a slightly weaker inequality.

**Lemma 8.7.** *Consider any  $(a, z)$  produced by RAISEPRICE. For every client  $j$  the following holds:*

- *If  $j \in \mathcal{D} \setminus \mathcal{B}$ , then there exists a tight facility  $i$  such that  $(1 + \sqrt{\delta} + \epsilon)\bar{\alpha}_j \geq d(j, i) + \sqrt{\delta t_i}$ .*
- *There exists a tight facility  $i$  such that  $6\bar{\alpha}_j^{(0)} \geq d(j, i) + \sqrt{\delta t_i}$ .*

*Proof.* The proof is by induction on the well-ordered set (with respect to the natural order  $\leq$ )

$$R = \{0\} \cup \{\alpha_j\}_{j \in \mathcal{D} \setminus \mathcal{B}} \cup \{(1 + \epsilon)\alpha_j^{(0)}\}_{j \in \mathcal{D}}.$$

Specifically, we prove the following induction hypothesis: for  $r \in R$ ,

- (a) each client  $j \in \mathcal{D} \setminus \mathcal{B}$  with  $\alpha_j \leq r$  has a tight facility  $i$  such that  $(1 + \sqrt{\delta} + \epsilon)\bar{\alpha}_j \geq d(j, i) + \sqrt{\delta t_i}$ ;
- (b) each client  $j \in \mathcal{D}$  with  $(1 + \epsilon)\alpha_j^{(0)} \leq r$  has a tight facility  $i$  such that  $6\bar{\alpha}_j^{(0)} \geq d(j, i) + \sqrt{\delta t_i}$ .

The statement then follows from the above with  $r = \arg \max_{r \in R} r$ .

For the base case (when  $r = 0$ ), the claim is vacuous since there is no client  $j$  such that  $\alpha_j \leq 0$  or  $(1 + \epsilon)\alpha_j^{(0)} \leq 0$  (because every  $\alpha$ -value is at least 1 by Invariant 2). For the induction step, we assume that each client  $j \in \mathcal{D} \setminus \mathcal{B}$  with  $\alpha_j < r$  satisfies (a) and each client  $j \in \mathcal{D}$  with  $(1 + \epsilon)\alpha_j^{(0)} < r$  satisfies (b). We need to prove that any client  $j_0 \in \mathcal{D} \setminus \mathcal{B}$  with  $\alpha_{j_0} = r$  (respectively,  $j_0 \in \mathcal{D}$  with  $(1 + \epsilon)\alpha_{j_0}^{(0)} = r$ ) satisfies (a) (respectively, (b)). We divide the proof into two cases.

**Case 1:**  $j_0 \in \mathcal{D} \setminus \mathcal{B}$  with  $\alpha_{j_0} = r$ . We prove that in this case  $j_0$  satisfies (a). Since  $j_0 \notin \mathcal{B}$ , either  $j_0$  has a witness,  $j_0$  is currently stopped, or there is another client  $j$  such that  $2\bar{\alpha}_{j_0} \geq d(j_0, j) + 6\bar{\alpha}_j^{(0)}$ .

Suppose first that  $j_0$  has a witness  $i$ . Then,  $i$  is a tight facility and, since  $j_0$  has a tight edge to  $i$ ,  $d(j_0, i) \leq \bar{\alpha}_{j_0}$ . Moreover,  $B(\alpha_{j_0}) \geq B(t_i)$  which implies that  $(1 + \frac{\epsilon}{2})\bar{\alpha}_{j_0} \geq \sqrt{(1 + \epsilon)\bar{\alpha}_{j_0}} \geq \sqrt{t_i}$ . Therefore, using that  $\sqrt{\delta} \leq 2$ ,

$$d(j_0, i) + \sqrt{\delta t_i} \leq (1 + \sqrt{\delta} + \epsilon)\bar{\alpha}_{j_0}.$$

Now suppose that  $j_0$  is stopped by another client  $j$ . Then  $\alpha_j \leq \alpha_{j_0}/3^2 = r/9$ . On the one hand, if  $j \in \mathcal{D} \setminus \mathcal{B}$ , we have  $d(j_0, i) + \sqrt{\delta t_i} \leq (1 + \sqrt{\delta} + \epsilon)\bar{\alpha}_j \leq 6\bar{\alpha}_j$  for some tight facility  $i$  by the induction hypothesis (a). On the other hand, if  $j \in \mathcal{B}$  then  $j$  is undecided so by Lemma 8.4,  $\alpha_j^{(0)} \leq \alpha_j$ . This in turn implies that  $\alpha_j^{(0)} \leq \alpha_j \leq r/9 < r/(1 + \epsilon)$ . We can thus apply the induction hypothesis (b) to  $j$ , to conclude that there is a tight facility  $i$  such that  $d(j, i) + \sqrt{\delta t_i} \leq 6\bar{\alpha}_j^{(0)} \leq 6\bar{\alpha}_j$ . From above we have that, whether  $j$  is in  $\mathcal{B}$  or not, there is a tight facility  $i$  such that

$$\begin{aligned} d(j_0, i) + \sqrt{\delta t_i} &\leq d(j_0, j) + d(j, i) + \sqrt{\delta t_i} \\ &\leq d(j_0, j) + 6\bar{\alpha}_j \\ &\leq 2\bar{\alpha}_{j_0} \leq (1 + \sqrt{\delta} + \epsilon)\bar{\alpha}_{j_0}, \end{aligned}$$

where the penultimate inequality uses the fact that  $j_0$  is stopped by  $j$  and thus  $2\bar{\alpha}_{j_0} \geq d(j_0, j) + 6\bar{\alpha}_j$ .

Finally, suppose that  $j_0$  is not stopped or witnessed. Then,  $j_0$  is currently undecided and, as  $j_0 \notin \mathcal{B}$ , there is a client  $j$  such that  $2\bar{\alpha}_{j_0} \geq d(j_0, j) + 6\bar{\alpha}_j^{(0)}$ . This implies that  $\alpha_j^{(0)} \leq \alpha_{j_0}/9 = r/9 < r/(1 + \epsilon)$ . We can thus apply the induction hypothesis (b) to  $j$  to conclude, that there is a tight facility  $i$  such that  $d(j, i) + \sqrt{\delta t_i} \leq 6\bar{\alpha}_j^{(0)}$ . Now, we have:

$$\begin{aligned} d(j_0, i) + \sqrt{\delta t_i} &\leq d(j_0, j) + d(j, i) + \sqrt{\delta t_i} \\ &\leq d(j_0, j) + 6\bar{\alpha}_j^{(0)} \\ &\leq 2\bar{\alpha}_{j_0} \leq (1 + \sqrt{\delta} + \epsilon)\bar{\alpha}_{j_0}. \end{aligned}$$

**Case 2:**  $j_0 \in \mathcal{D}$  **with**  $(1 + \epsilon)\alpha_{j_0}^{(0)} = r$ . We prove that in this case  $j_0$  satisfies (b). Suppose first that  $\alpha_{j_0} < \alpha_{j_0}^{(0)}$ . Then  $j_0$  is decided by Lemma 8.4. Therefore  $j_0 \in \mathcal{D} \setminus \mathcal{B}$  with  $\alpha_{j_0} < r$  and so by the induction hypothesis (a) there is a tight facility  $i$  satisfying  $d(j_0, i) + \sqrt{\delta t_i} \leq (1 + \sqrt{\delta} + \epsilon)\bar{\alpha}_{j_0} < 6\bar{\alpha}_{j_0}^{(0)}$ , as required. Similarly, if  $j_0 \in U^{(0)}$  then by Corollary 8.6,  $j_0$  is decided and so

$$\alpha_{j_0} \leq \alpha_{j_0}^{(0)} + \epsilon_z < (1 + \epsilon)\alpha_{j_0}^{(0)} = r,$$

where the second inequality follows from  $\epsilon_z < \epsilon$  and  $\alpha_{j_0}^{(0)} \geq 1$  by Invariant 2. We can thus again apply the induction hypothesis (a) to conclude that there is a tight facility  $i$  satisfying  $d(j_0, i) + \sqrt{\delta t_i} \leq (1 + \sqrt{\delta} + \epsilon)\bar{\alpha}_{j_0} \leq 6\bar{\alpha}_{j_0}^{(0)}$ . Thus, from now on, we assume that  $\alpha_{j_0} \geq \alpha_{j_0}^{(0)}$  and that  $j_0 \notin U^{(0)}$ . We divide the remaining part of the analysis into two sub-cases depending on whether  $j_0$  was stopped in  $\alpha^{(0)}$ .

First, suppose that  $j_0$  was stopped in  $\alpha^{(0)}$  by another client  $j$ . Then  $\alpha_j^{(0)} \leq \alpha_{j_0}^{(0)}/9 < r/(1 + \epsilon)$  and so by the induction hypothesis (b), there is a tight facility  $i$  satisfying  $d(j, i) + \sqrt{\delta t_i} \leq 6\bar{\alpha}_j^{(0)}$ . Hence,

$$\begin{aligned} d(j_0, i) + \sqrt{\delta t_i} &\leq d(j_0, j) + d(j, i) + \sqrt{\delta t_i} \\ &\leq d(j_0, j) + 6\bar{\alpha}_j^{(0)} \\ &\leq 2\bar{\alpha}_{j_0}^{(0)} < 6\bar{\alpha}_{j_0}^{(0)}. \end{aligned}$$

Finally, suppose that  $j_0$  was not stopped in  $\alpha^{(0)}$ . Then since every client is decided in  $\alpha^{(0)}$  (Invariant 4)  $j_0$  had a witness  $i$  in  $\alpha^{(0)}$ . Moreover, as  $j_0 \notin U^{(0)}$ , we may assume that  $i \neq i^+$  and so  $z_i = z_i^{(0)}$ . By the definition of a witness,  $\alpha_{j_1}^{(0)} \leq (1 + \epsilon)\alpha_{j_0}^{(0)}$  for all  $j_1 \in N^{(0)}(i)$ . If  $\alpha_{j_1} \geq \alpha_{j_1}^{(0)}$  for all  $j_1 \in N^{(0)}(i)$ , then, since  $z_i = z_i^{(0)}$ , our feasibility invariant (Invariant 2) implies that in fact  $\alpha_{j_1} = \alpha_{j_1}^{(0)}$  for all  $j_1 \in N^{(0)}(i)$  and so  $N(i) = N^{(0)}(i)$ . Therefore, in this case  $i$  is still a witness for  $j_0$  and  $d(j_0, i) + \sqrt{\delta t_i} \leq (1 + \sqrt{\delta} + \epsilon)\bar{\alpha}_{j_0}^{(0)} \leq 6\bar{\alpha}_{j_0}^{(0)}$ . It remains to consider the case when  $\alpha_{j_1} < \alpha_{j_1}^{(0)}$  for some  $j_1 \in N^{(0)}(i)$  (note that  $j_1 \neq j_0$ , since  $\alpha_{j_0} \geq \alpha_{j_0}^{(0)}$  by assumption). Since  $\alpha_{j_1} < \alpha_{j_1}^{(0)}$ ,  $j_1$  must be decided (by Lemma 8.4) and so  $j_1 \in \mathcal{D} \setminus \mathcal{B}$ . Moreover,  $\alpha_{j_1} < \alpha_{j_1}^{(0)} \leq (1 + \epsilon)\alpha_{j_0}^{(0)} = r$ , and so we can apply the induction hypothesis (a) to conclude that there is a tight facility  $i_1$  satisfying  $d(j_1, i_1) + \sqrt{\delta t_{i_1}} \leq (1 + \sqrt{\delta} + \epsilon)\bar{\alpha}_{j_1} < (1 + \sqrt{\delta} + \epsilon)\bar{\alpha}_{j_1}^{(0)}$ . Then,

$$\begin{aligned} d(j_0, i_1) + \sqrt{\delta t_{i_1}} &\leq d(j_0, i) + d(i, j_1) + d(j_1, i_1) + \sqrt{\delta t_{i_1}} \\ &< \bar{\alpha}_{j_0}^{(0)} + \bar{\alpha}_{j_1}^{(0)} + (1 + \sqrt{\delta} + \epsilon)\bar{\alpha}_{j_1}^{(0)} \\ &\leq \bar{\alpha}_{j_0}^{(0)} + (1 + \epsilon)\bar{\alpha}_{j_0}^{(0)} + (1 + \epsilon)(1 + \sqrt{\delta} + \epsilon)\bar{\alpha}_{j_0}^{(0)} \\ &\leq 6\bar{\alpha}_{j_0}^{(0)}, \end{aligned}$$

as required.  $\square$

Lemma 8.7 shows that the clients in  $\mathcal{D} \setminus \mathcal{B}$  satisfy the first inequality of Property 2 in Definition 5.1 while the potentially bad clients  $j \in \mathcal{B}$  satisfy a slightly weaker inequality. It remains to prove that the potentially bad clients will have a small contribution towards the total cost of our solution.

## 8.4 Showing that $\alpha$ -values are stable

The key to our remaining analysis is showing that the  $\alpha$ -values are relatively well-behaved throughout the algorithm. The following lemma implies that SWEEP decreases an  $\alpha_{j'}$  only because it is increasing an  $\alpha_j$  which is at most a constant factor smaller. This will imply the required stability properties.

**Lemma 8.8.** *At any time during Algorithm 1: if a client  $j$  has a tight edge to some facility, then  $\alpha_{j'} \leq 19^2 \alpha_j$  for every other client  $j'$  with a tight edge to this facility.*

*Proof.* We prove the following stronger statement: at any time during Algorithm 1, we have

$$2\bar{\alpha}_{j'} \leq d(j', j) + 18\bar{\alpha}_j \quad (8.2)$$

for any pair  $j, j'$  of clients. To see that this implies the lemma consider two clients  $j$  and  $j'$  that both have tight edges to  $i^*$ . Then

$$2\bar{\alpha}_{j'} \leq d(j', j) + 18\bar{\alpha}_j \leq d(j', i^*) + d(i^*, j) + 18\bar{\alpha}_j \leq \bar{\alpha}_{j'} + \bar{\alpha}_j + 18\bar{\alpha}_j,$$

which implies that  $\alpha_{j'} \leq 19^2 \alpha_j$ .

Inequality (8.2) is clearly satisfied by the initial solution  $\alpha^{(0)}$  constructed at the beginning of Algorithm 1, since we stop increasing any  $\alpha_j$  as soon as  $2\bar{\alpha}_j \geq d(j', j) + 6\bar{\alpha}_{j'}$  for any client  $j'$ , and neither  $\alpha_j$  nor  $\alpha_{j'}$  are later changed. We now show that (8.2) continues to hold throughout the execution of Algorithm 1. The only procedure that updates the dual solution is SWEEP, so let us analyze its behavior.

First note that the inequality cannot become violated by increasing  $j'$ , because as soon as  $2\bar{\alpha}_{j'} \geq d(j', j) + 6\bar{\alpha}_j$ ,  $j'$  will be removed from  $A$  by Rule 2 of SWEEP. It remains to prove that the inequality does not become violated because  $j$  is decreasing. To this end, consider a time when  $j$  is decreasing. Then, by the definition of SWEEP, there must be some potentially tight facility  $i$ , such that  $j \in N(i)$  with  $\alpha_j = t_i$ . Since  $j$  has the largest  $\alpha$ -value in  $N(i)$  and  $i$  is potentially tight, there is some client  $j_1 \in N(i)$  (note that possibly  $j_1 = j$ ) such that  $\alpha_{j_1}^{(0)} \leq \alpha_{j_1} \leq \alpha_j$ . We show the following:

**Claim.** *There exists some facility  $i^*$  such that  $i^*$  was tight in  $(\alpha^{(0)}, z^{(0)})$  and also:*

$$d(j_1, i^*) \leq 2\bar{\alpha}_{j_1}^{(0)} \leq 2\bar{\alpha}_j \quad \text{and} \quad \alpha_{j''}^{(0)} \leq (1 + \epsilon)\alpha_{j_1}^{(0)} \leq (1 + \epsilon)\alpha_j \text{ for all } j'' \in N^{(0)}(i^*).$$

*Proof.* By Invariant 4, every client must be decided in  $(\alpha^{(0)}, z^{(0)})$ . Consider client  $j_1$ . If  $j_1$  was witnessed in  $(\alpha^{(0)}, z^{(0)})$ , then there was a tight facility  $i^*$  such that  $d(j_1, i^*) \leq \bar{\alpha}_{j_1}^{(0)} \leq \bar{\alpha}_j$  and  $\alpha_{j''}^{(0)} \leq (1 + \epsilon)\alpha_{j_1}^{(0)}$  for every  $j'' \in N^{(0)}(i^*)$ . If  $j_1$  was stopped by a client  $j_2$  in  $(\alpha^{(0)}, z^{(0)})$  (i.e.,  $2\bar{\alpha}_{j_1}^{(0)} \geq d(j_1, j_2) + 6\bar{\alpha}_{j_2}^{(0)}$ ), then we may assume that  $j_2$  is witnessed by Lemma 7.1. In this case, let  $i^*$  be the witness of  $j_2$ . Then,

$$d(j_1, i^*) \leq d(j_1, j_2) + d(j_2, i^*) \leq d(j_1, j_2) + \bar{\alpha}_{j_2}^{(0)} \leq 2\bar{\alpha}_{j_1}^{(0)} \leq 2\bar{\alpha}_j,$$

and also

$$\alpha_{j''}^{(0)} \leq (1 + \epsilon)\alpha_{j_2}^{(0)} \leq \alpha_{j_1}^{(0)} \leq \alpha_j,$$

for all  $j'' \in N^{(0)}(i^*)$ . In either case, the claim holds.  $\square$



Now, let  $i^*$  be the facility guaranteed to exist by the Claim. Consider the dual solution  $\alpha^{(p)}$  at the last time that  $j'$  was previously increased. Then, we must have  $\alpha_{j'}^{(p)} \geq \alpha_{j'}$ . Additionally, since Algorithm 1 never decreases any facility's price, and the current call to RAISEPRICE has increased any facility's price by at most  $\epsilon_z$ , we have  $z_{i^*}^{(p)} \leq z_{i^*} \leq z_{i^*}^{(0)} + \epsilon_z$ . Let  $j^* = \arg \min_{j'' \in N^{(0)}(i^*)} \alpha_{j''}^{(p)}$ . We claim that:

$$\alpha_{j^*}^{(p)} = \min_{j'' \in N^{(0)}(i^*)} \alpha_{j''}^{(p)} \leq (1 + \epsilon)\alpha_j + \epsilon_z. \quad (8.3)$$

Indeed, otherwise by the Claim, we would have  $\alpha_{j''}^{(p)} > (1 + \epsilon)\alpha_j + \epsilon_z \geq \alpha_{j''}^{(0)} + \epsilon_z$  for every  $j'' \in N^{(0)}(i^*)$ . Then, since  $i^*$  is tight in  $(\alpha^{(0)}, z^{(0)})$  we would have:

$$\begin{aligned} \sum_{j'' \in \mathcal{D}} [\alpha_{j''}^{(p)} - d(j'', i^*)]^+ &\geq \sum_{j'' \in N^{(0)}(i^*)} [\alpha_{j''}^{(p)} - d(j'', i^*)]^+ \\ &> \sum_{j'' \in N^{(0)}(i^*)} [\alpha_{j''}^{(0)} + \epsilon_z - d(j'', i^*)]^+ \geq z_{i^*}^{(0)} + \epsilon_z \geq z^{(p)}, \end{aligned}$$

contradicting Invariant 2.

We shall now show that (8.3) and the claim imply (8.2). Since  $j'$  was increasing when  $\alpha^{(p)}$  was maintained, Rule 2 of SWEEP implies that:

$$\begin{aligned} 2\bar{\alpha}_{j'} &\leq 2\bar{\alpha}_{j'}^{(p)} \leq d(j', j^*) + 6\bar{\alpha}_{j^*}^{(p)} && (j' \text{ was last increased in } \alpha^{(p)}) \\ &\leq d(j', j) + d(j, j_1) + d(j_1, i^*) + d(i^*, j^*) + 6\bar{\alpha}_{j^*}^{(p)} && (\text{triangle inequality}) \\ &\leq d(j', j) + d(j, j_1) + d(j_1, i^*) + \bar{\alpha}_{j^*}^{(0)} + 6\bar{\alpha}_{j^*}^{(p)} && (j^* \in N^{(0)}(i^*)) \\ &\leq d(j', j) + d(j, j_1) + d(j_1, i^*) + \bar{\alpha}_{j^*}^{(0)} + 12\bar{\alpha}_j && (\text{inequality (8.3)}) \\ &\leq d(j', j) + d(j, j_1) + 2\bar{\alpha}_j + (1 + \epsilon)^{1/2}\bar{\alpha}_j + 12\bar{\alpha}_j && (j^* \in N^{(0)}(i^*) \text{ and Claim above}) \\ &\leq d(j', j) + d(j, j_1) + 16\bar{\alpha}_j && (\text{arithmetic}) \\ &\leq d(j', j) + d(j, i) + d(i, j_1) + 16\bar{\alpha}_j && (\text{triangle inequality}) \\ &\leq d(j', j) + \bar{\alpha}_j + \bar{\alpha}_{j_1} + 16\bar{\alpha}_j && (j, j_1 \in N(i)) \\ &\leq d(j', j) + 18\bar{\alpha}_j. && (\alpha_j \geq \alpha_{j_1} \text{ since } j \text{ decreasing}) \end{aligned}$$

and thus (8.2) remains satisfied when  $j$  is decreasing.  $\square$

Using Lemma 8.8, we can now prove that RAISEPRICE produces a close sequence of solutions, and also bound the total number of clients in  $\mathcal{B}$  for any solution produced by RAISEPRICE. For both of these tasks, we make use of the following auxiliary lemma, which is a consequence of Lemma 8.8.

**Lemma 8.9.** *Throughout stage  $s$ ,  $\alpha_j \leq \alpha_j^{(0)}$  for all  $j$  with  $\alpha_j > \theta_s$  and if  $\alpha_j^{(0)} \geq 20^2\theta_s$  or  $\alpha_j \geq 20^2\theta_s$  then  $\alpha_j = \alpha_j^{(0)}$  for all  $j$ .*

*Proof.* For the first claim, we show that any client  $\alpha_j$  with  $\alpha_j \geq \theta_s$  can continue to increase in stage  $s$  only while  $\alpha_j < \alpha_j^{(0)}$ . Indeed, if  $\alpha_j \geq \theta_s$  then once  $\alpha_j = \alpha_j^{(0)}$ ,  $j$  will immediately be removed from  $A$  by Rule 4.

For the remaining claim, suppose first that  $\alpha_j^{(0)} \geq 20^2\theta_s$ . Suppose further, towards contradiction, that  $\alpha_j < \alpha_j^{(0)}$  at some moment in stage  $s$  or earlier, and let  $\alpha^{(-)}$  be the value of  $\alpha$  at this time. Then, at some moment in stage  $s$  or earlier, we must have had  $\alpha_j^{(-)} < \alpha_j < \alpha_j^{(0)}$ , and  $\alpha_j \geq 19^2\theta_s$  but  $j$  decreasing. Since  $j$  is being decreased by SWEEP at this moment, we must have  $j \in N(i)$  for

a potentially tight facility  $i$ . Since  $\alpha_j^{(0)} > \alpha_j$  we must also have  $j \in N^{(0)}(i)$ . However, Lemma 8.8 implies that for every other  $j' \in N(i)$  at this moment we have  $\alpha_{j'} \geq 19^{-2}\alpha_j \geq \theta_s$ . Thus, by the first claim,  $\alpha_{j'} \leq \alpha_{j'}^{(0)}$  for all  $j' \in N(i)$ . This contradicts the fact that  $i$  is potentially tight, since  $j \in N^{(0)}(i)$  with  $\alpha_j < \alpha_j^{(0)}$ .

Finally, suppose that  $\alpha_j \geq 20^2\theta_s$ . Then, by the first claim, we must have  $\alpha_j \leq \alpha_j^{(0)}$  and so also  $\alpha_j^{(0)} \geq 20^2\theta_s$ . Then, as we have just shown,  $\alpha_j = \alpha_j^{(0)}$ .  $\square$

#### 8.4.1 RAISEPRICE produces a close sequence of $\alpha$ -values in polynomial time

In the preceding section, we showed that all of the  $\alpha$ -values are relatively stable throughout the algorithm. Using those observations, we can now prove that RAISEPRICE indeed produces a close sequence of  $\alpha$ -values. To that end, let us select the remaining parameters  $K$ ,  $\sigma$ , and  $\epsilon_z$  used in RAISEPRICE.

Recall that the thresholds used by RAISEPRICE are defined by:

$$\theta_1 = (\max_{j \in U^{(0)}} \alpha_j^{(0)} + 2\epsilon_z)(1 + \epsilon)^\sigma \quad \text{and} \quad \theta_s = (1 + \epsilon)^K \theta_{s-1} \text{ for all } s > 1.$$

Therefore, the ratio of two consecutive thresholds is  $\theta_s/\theta_{s-1} = (1 + \epsilon)^K$ . We select  $K$  to be the smallest integer satisfying

$$(1 + \epsilon)^K \geq C_0^{2/\gamma^4}, \quad \text{where } C_0 = 81 \cdot 25 \cdot 20^8.$$

Note that  $K = \Theta(\epsilon^{-1}\gamma^{-4})$ . Given  $K$ , we select an integer “shift”  $\sigma$  uniformly at random from the interval  $(0, K/2]$ . This completes the definition of our thresholds. Finally, we set the price increment  $\epsilon_z$  to:

$$\epsilon_z = n^{-6(K+C_1+2)-3} \quad \text{where } C_1 = \lceil \log_{1+\epsilon}(20^4) \rceil + 1 = O(\epsilon^{-1}). \quad (8.4)$$

Using these parameters, we can show that the sequence of solutions  $(\alpha, z)$  produced by RAISEPRICE is indeed close. Because each successive  $\alpha$ -value in this sequence is produced by calling SWEEP on the previous value, it suffices to show the following.

**Proposition 8.10.** *Each call to SWEEP changes every  $\alpha_j$  by at most  $n^{-2}$ .*

*Proof.* Consider a call to SWEEP performed in stage  $s$ . By the definition of SWEEP, it suffices to bound how much  $\alpha_j$  has changed at the moment it is removed from  $A$ , since it is not subsequently changed. Let us begin by bounding how much any  $\alpha_j$  may be increased. As in our analysis of QUASISWEEP, it will then be possible to bound how much any  $\alpha$ -value is decreased. Let  $\alpha^{(1)}$  be the value of  $\alpha$  before this call to SWEEP, and let  $\mu = \min_{j \in U} \alpha_j^{(1)}$ . We first show the following:

**Claim.** *Any  $\alpha_j$  can increase by at most  $\epsilon_z n^{6(b+1)}$  while  $B(\theta) \leq B(\mu) + b$ .*

*Proof.* The proof is by induction on  $b = -1, 0, 1, \dots$

**Base case  $b = -1$ :** Let us first show that this base case indeed occurs in any call to SWEEP. Initially we have  $\theta = 0$  and, by Invariant 2,  $\mu \geq 1$ . Thus, at the start of any call to SWEEP, we must have  $B(\theta) = 0$  and  $B(\mu) \geq 1$ . Now, note that while  $B(\theta) \leq B(\mu) - 1$  we must have  $\theta < \mu$ . Then, by Property 3 of SWEEP no  $\alpha$ -value has yet been altered, and so the claim holds trivially.

**Inductive step ( $b \geq 0$ ):** Now suppose that some  $\alpha_j$  is increased by at least  $\epsilon_z$  while  $B(\theta) \leq B(\mu) + b$ . Otherwise, the claim is immediate since  $\epsilon_z < \epsilon_z n^{6(b+1)}$ . Note that while this  $\alpha_j$  is increasing we must also have  $\alpha_j = \theta$  and so  $B(\alpha_j) \leq B(\mu) + b$ .

First, suppose that  $\alpha_j < \alpha_j^{(1)}$ . Then,  $\alpha_j$  was previously decreased. Moreover, since  $\alpha_j$  was increased by at least  $\epsilon_z$  while  $B(\theta) \leq B(\mu) + b$ , we must have previously decreased  $\alpha_j$  while  $B(\alpha_j) \leq B(\mu) + b$ . In particular, at the last moment  $\alpha_j$  was decreased, we must have had  $B(\alpha_j) \leq B(\mu) + b$ , and since  $\alpha_j$  was decreasing at this moment, we also had  $B(\theta) < B(\alpha_j)$ . Therefore,  $\alpha_j$  was decreased only while  $B(\theta) < B(\mu) + b$ . Moreover, during this time,  $j$ 's  $\alpha$ -value was decreased at most  $|A| \leq n$  times the amount that any other client's  $\alpha$ -value was increased. By the induction hypothesis, any client's  $\alpha$ -value can increase at most  $\epsilon_z n^{6b}$  while  $B(\theta) < B(\mu) + b$ . Thus,  $\alpha_j$  has decreased at most  $\epsilon_z \cdot n^{6b+1}$ , and after increasing  $\alpha_j$  by at most this amount, we will again have  $\alpha_j = \alpha_j^{(1)}$ .

Next, let us bound how much  $j$ 's  $\alpha$ -value may increase while  $\alpha_j \geq \alpha_j^{(1)}$  (and still  $B(\alpha_j) \leq B(\mu) + b$ ). We now consider three cases, based on the initial status of  $j$  in  $\alpha^{(1)}$ .

If  $j$  is undecided initially, then  $j \in U$  and  $\alpha_j$  can increase by at most  $\epsilon_z \leq \epsilon_z n^{6b}$  (since  $b \geq 0$ ) before it is removed by Rule 3.

Next, suppose that  $j$  had some witness  $i$  in  $\alpha^{(1)}$ , and let  $N^{(1)}(i)$  be the set of clients paying for  $i$  in  $\alpha^{(1)}$ . For each  $j' \in N^{(1)}(i)$  we must have  $B(\alpha_{j'}^{(1)}) \leq B(\alpha_j^{(1)}) \leq B(\mu) + b$ , and so  $\alpha_{j'}$  is decreased by SWEEP only while  $B(\theta) \leq B(\mu) + b - 1$ . By the same argument given above (when considering the case that  $\alpha_j < \alpha_j^{(1)}$ ), the  $\alpha$ -value of any such  $j' \in N^{(1)}(i)$  can decrease at most  $\epsilon_z n^{6b+1}$  during SWEEP. Thus, the total contribution to  $i$  can decrease at most  $n \cdot \epsilon_z n^{6b+1} = \epsilon_z n^{6b+2}$  during SWEEP. After increasing  $\alpha_j$  by at most this amount,  $i$  will again be tight. Moreover, at this moment any client  $j'$  contributing to  $i$  was either already added to  $A$  (and potentially also removed), in which case  $B(\alpha_{j'}) \leq B(\theta) = B(\alpha_j)$ , or it was not already added to  $A$ , in which case  $B(\alpha_{j'}) \leq B(\alpha_j^{(1)}) \leq B(\alpha_j^{(1)}) \leq B(\alpha_j)$ . Thus, at this moment  $i$  is a witness for  $j$ , and so  $j$  will be removed from  $A$  by Rule 1.

Finally, suppose that  $j$  was initially stopped by some client  $j'$ . Then, by Lemma 7.1, we may assume that  $j'$  was *not* stopped. Let  $\Delta = \bar{\alpha}_{j'} - \bar{\alpha}_{j'}^{(1)}$  be the amount that  $\bar{\alpha}_{j'}$  has been increased by SWEEP. Then, once  $\bar{\alpha}_j - \bar{\alpha}_j^{(1)} \geq 3\Delta$ , we will have:

$$2\bar{\alpha}_j \geq 2\bar{\alpha}_j^{(1)} + 6(\bar{\alpha}_{j'} - \bar{\alpha}_{j'}^{(1)}) \geq d(j, j') + 6\bar{\alpha}_{j'},$$

where in the last inequality we have used the fact that  $j'$  stopped  $j$  in  $\alpha^{(1)}$ . Thus,  $\bar{\alpha}_j$  can increase by at most  $3\Delta$ , before  $j$  will again be stopped by  $j'$  and removed from  $A$  by Rule 2. It remains to bound the corresponding increases in  $\alpha_j$  and  $\alpha_{j'}$ . We have:

$$\alpha_j - \alpha_j^{(1)} \leq \left( \bar{\alpha}_j^{(1)} + 3\Delta \right)^2 - \alpha_j^{(1)} = 6\Delta \cdot \bar{\alpha}_j^{(1)} + 9\Delta^2.$$

Now, let us bound the right hand side. Since  $j'$  is not stopped, the previous cases show that  $\alpha_{j'} - \alpha_{j'}^{(1)} \leq \epsilon_z n^{6b+2}$ . Then, we have:

$$\Delta^2 = \left( \sqrt{\alpha_{j'}} - \sqrt{\alpha_{j'}^{(1)}} \right)^2 \leq \left( \sqrt{\alpha_{j'}^{(1)} + \epsilon_z n^{6b+2}} - \sqrt{\alpha_{j'}^{(1)}} \right)^2 \leq \epsilon_z n^{6b+2},$$

where the last inequality follows from  $\sqrt{a+b} \leq \sqrt{a} + \sqrt{b}$  for all  $a, b \in \mathbb{R}_+$ . On the other hand,

since  $g(x) = \sqrt{x}$  is a concave function of  $x$  for all  $x > 0$ , we have:

$$\Delta = \sqrt{\alpha_{j'}} - \sqrt{\alpha_{j'}^{(1)}} \leq g(\alpha_{j'}^{(1)} + \epsilon_z n^{6b+2}) - g(\alpha_{j'}^{(1)}) \leq \epsilon_z n^{6b+2} \cdot g'(\alpha_{j'}^{(1)}) = \frac{\epsilon_z n^{6b+2}}{2\sqrt{\alpha_{j'}^{(1)}}} \leq \frac{\epsilon_z n^{6b+2}}{2},$$

where the last inequality follows from Invariant 2, which implies  $\alpha_{j'}^{(1)} \geq 1$ . Combining the above bounds, in this case we have

$$\begin{aligned} \alpha_j - \alpha_j^{(1)} &\leq 6\Delta \cdot \bar{\alpha}_j^{(1)} + 9\Delta^2 \leq 6\bar{\alpha}_j^{(1)} \cdot \frac{\epsilon_z n^{6b+2}}{2} + 9\epsilon_z n^{6b+2} \\ &\leq 3\sqrt{5n^7} \cdot \epsilon_z n^{6b+2} + 9\epsilon_z n^{6b+2} \leq 9\epsilon_z n^{6b+11/2}, \end{aligned}$$

where the penultimate inequality follows from the feasibility invariant (Invariant 2) and the pre-processing of Lemma 4.1 (that all squared-distances are at most  $n^6$ ) which together imply that  $\alpha_j \leq \min_i (z_i + d(i, j)^2) \leq 4n^7 + n^6 \leq 5n^7$  for all  $j \in D$ .

Combining all of the above cases,  $\alpha_j$  can increase at most  $\epsilon_z n^{6b+1}$ , until  $\alpha_j = \alpha_j^{(1)}$  and then at most an additional  $9\epsilon_z n^{6b+11/2}$ . Thus, the total increase in  $\alpha_j$  while  $B(\theta) \leq B(\mu) + b$  is at most  $9\epsilon_z n^{6b+11/2} + \epsilon_z n^{6b+1} \leq \epsilon_z n^{6b+6}$ , as required.  $\square$

We now complete the proof of Proposition 8.10. By Lemma 8.9, no  $\alpha_j \geq 20^2 \theta_s$  is changed by SWEEP in any stage  $s$ , and so once  $B(\theta) \geq B(20^2 \theta_s)$  no  $\alpha$ -values are changed. By the Claim, we then have that in any call to SWEEP in stage  $s$ , each client's  $\alpha$ -value is increased at most  $\epsilon_z n^{6(b+1)}$  where

$$b = B(20^2 \theta_s) - B(\mu) \leq \lfloor \log_{1+\epsilon}(20^2 \theta_s / \mu) \rfloor + 1.$$

We now bound the above value  $b$  for every stage  $s$ .

In stage 1, we execute only a single call to SWEEP (as shown in Lemma 8.3) and in this call,  $\mu = \min_{j \in U^{(0)}} \alpha_j^{(0)}$ . Since every  $j \in U^{(0)}$  must have a tight edge to the facility  $i^+$  in  $\alpha^{(0)}$ , Lemma 8.8 implies that  $\nu \triangleq \max_{j \in U^{(0)}} \alpha_j^{(0)} \leq 20^2 \mu$ . Then, recall that

$$\theta_1 = (\nu + 2\epsilon_z)(1 + \epsilon)^\sigma \leq \nu(1 + \epsilon)^K \leq 20^2 \mu(1 + \epsilon)^K,$$

where we have used that  $\nu \geq 1$  (by Invariant 2),  $\epsilon_z < \epsilon$  and  $\sigma \leq K/2 < K$ . Finally, recalling that  $C_1 = \lceil \log_{1+\epsilon}(20^4) \rceil$ , we have:

$$b \leq \log_{1+\epsilon}(20^2 \theta_1 / \mu) + 1 \leq \log_{1+\epsilon}(20^2 \cdot 20^2 \cdot (1 + \epsilon)^K) + 1 \leq K + C_1 + 1.$$

In stage  $s > 1$ , we have  $\mu \geq \theta_{s-1}$  by Lemma 8.5. Then, recall that  $\theta_s = (1 + \epsilon)^K \theta_{s-1}$ . Then, we have:

$$b \leq \log_{1+\epsilon}(20^2 \theta_s / \mu) + 1 \leq \log_{1+\epsilon}(20^2 (1 + \epsilon)^K) + 1 < K + C_1 + 1.$$

In any case, the maximum increase in any client's  $\alpha$ -value is at most  $\epsilon_z n^{6(K+C_1+2)} = n^{-3}$  (recalling that by definition  $\epsilon_z = n^{-6(K+C_1+2)-3}$ ). As we have already observed above in the proof of the Claim, each  $\alpha$ -value can decrease at most  $n$  times this amount. Thus, no  $\alpha$ -value can decrease more than  $n^{-2}$ .  $\square$

### 8.4.2 Bounding the number of clients in $\mathcal{B}$

We bound the number of clients in  $\mathcal{B}$  by showing that such clients need to have an  $\alpha^{(0)}$ -value close to a threshold  $\theta_s$ . We then select the thresholds so that only a tiny fraction of the clients can be in  $\mathcal{B}$ .

**Lemma 8.11.** *Suppose that  $j \in \mathcal{B}$  for some  $(\alpha, z)$  produced by RAISEPRICE. Then, we must have  $\frac{1}{81}\theta_s \leq \alpha_j^{(0)} < 25 \cdot 20^4\theta_s$  for some  $s$ .*

*Proof.* Consider a call to RAISEPRICE and let  $(\alpha^{(0)}, z^{(0)})$  be the input solution. We denote by  $(\alpha^{(\ell)}, z^{(\ell)})$  the solution produced by the  $\ell$ th call to SWEEP in the execution of RAISEPRICE. We also use  $\mathcal{B}^{(\ell)}$  to refer to the set  $\mathcal{B}$  of potentially bad clients associated to the solution  $(\alpha^{(\ell)}, z^{(\ell)})$ . That is,

$$\mathcal{B}^{(\ell)} = \{j \in \mathcal{D} : j \text{ is undecided in } (\alpha^{(\ell)}, z^{(\ell)}) \text{ and } 2\bar{\alpha}_j^{(\ell)} < d(j, j') + 6\bar{\alpha}_{j'}^{(0)} \text{ for all clients } j' \in \mathcal{D}\}.$$

Note that  $\mathcal{B}^{(0)} = \emptyset$  since every client is decided in  $(\alpha^{(0)}, z^{(0)})$  by Invariant 4. We prove the lemma by showing the following claim by induction on  $\ell$ :

**Claim.** *For each client  $j \in \mathcal{B}^{(\ell)}$  there is a client  $j'$  such that  $\bar{\alpha}_j^{(\ell)} \geq d(j, j') + \bar{\alpha}_{j'}^{(0)}$  and  $\frac{1}{9}\theta_s \leq \alpha_{j'}^{(0)} \leq 20^4\theta_s$ , for some  $s$ .*

Before proving the claim, let us show that it indeed implies the Lemma. Suppose that  $j \in \mathcal{B}^{(\ell)}$  for some  $\ell$ . Then, selecting  $j' = j$  in the definition of  $\mathcal{B}^{(\ell)}$ , we must have  $2\bar{\alpha}_j^{(\ell)} < 6\bar{\alpha}_j^{(0)}$  and so  $\alpha_j^{(\ell)} < 9\alpha_j^{(0)}$ . Now, consider the client  $j'$  and stage  $s$  guaranteed by the claim. Then, we must have:

$$\alpha_j^{(0)} \geq \frac{1}{9}\alpha_j^{(\ell)} \geq \frac{1}{9}\alpha_{j'}^{(0)} \geq \frac{1}{81}\theta_s.$$

Moreover, because  $j \in \mathcal{B}^{(\ell)}$ ,  $j$  is undecided in  $(\alpha^{(\ell)}, z^{(\ell)})$  and so by Lemma 8.4,  $\alpha_j^{(0)} \leq \alpha_j^{(\ell)}$ . Also, we must have:

$$2\bar{\alpha}_j^{(\ell)} < d(j, j') + 6\bar{\alpha}_{j'}^{(0)} \leq d(j, j') + \bar{\alpha}_{j'}^{(0)} + \sqrt{25 \cdot 20^4 \cdot \theta_s} \leq \bar{\alpha}_j^{(\ell)} + \sqrt{25 \cdot 20^4 \cdot \theta_s}.$$

Thus,  $\alpha_j^{(0)} \leq \alpha_j^{(\ell)} \leq 25 \cdot 20^4\theta_s$ , as well. □

*Proof of the Claim.* The base case when  $\ell = 0$  is trivially true since  $\mathcal{B}^{(0)} = \emptyset$ . For the inductive step, we assume the induction hypothesis for  $0, 1, \dots, \ell - 1$  and prove it for  $\ell$ . Any client  $j \in \mathcal{B}^{(\ell)}$  is undecided and (by Property 1 of SWEEP) must have been removed from  $A$  by one of the Rules 3, 4, or 5. We divide the analysis based on these three cases.

**Case 1:  $j \in \mathcal{B}^{(\ell)}$  was removed by Rule 3.** In this case, we must have  $j \in U$ , by the definition of Rule 3. Moreover, since all clients in  $U^{(0)}$  are decided in every solution produced by RAISEPRICE (Corollary 8.6), we must have that  $\ell \geq 2$ . Thus,  $j$  must have been undecided in the previously produced solution  $(\alpha^{(\ell-1)}, z^{(\ell-1)})$ , and  $\alpha_j^{(\ell)} = \alpha_j^{(\ell-1)} + \epsilon_z$ . Then,  $j \in \mathcal{B}^{(\ell-1)}$ , as well, since

$$2\bar{\alpha}_j^{(\ell-1)} < 2\bar{\alpha}_j^{(\ell)} < d(j, j') + 6\bar{\alpha}_{j'}^{(0)} \quad \text{for all } j' \in \mathcal{D}.$$

The statement then follows from the induction hypothesis and from that  $\alpha_j^{(\ell)} \geq \alpha_j^{(\ell-1)}$ .

**Case 2:  $j \in \mathcal{B}^{(\ell)}$  was removed by Rule 4.** By the definition of Rule 4, we must have  $\alpha_j^{(\ell)} \geq \alpha_j^{(0)}$  and  $\alpha_j^{(\ell)} \geq \theta_s$ , where  $s$  is the stage in which  $\alpha^{(\ell)}$  was produced. In this case, we prove the claim for  $j' = j$ . Clearly, we have  $\bar{\alpha}_j^{(\ell)} \geq \bar{\alpha}_j^{(0)} = d(j, j) + \bar{\alpha}_j^{(0)}$ . Next, we prove that  $\frac{1}{9}\theta_s \leq \alpha_j^{(0)} \leq 20^4\theta_s$ . For the lower bound, we observe that since  $j \in \mathcal{B}^{(\ell)}$  we must have  $2\bar{\alpha}_j^{(\ell)} < 6\bar{\alpha}_j^{(0)}$  and so  $\frac{1}{9}\theta_s \leq \frac{1}{9}\alpha_j^{(\ell)} \leq \alpha_j^{(0)}$ . We now prove the upper bound. First, note that  $j$  was not stopped by any client  $j'$  in  $(\alpha^{(0)}, z^{(0)})$  since then we would have  $2\bar{\alpha}_j^{(\ell)} \geq 2\bar{\alpha}_j^{(0)} \geq d(j, j') + 6\bar{\alpha}_{j'}^{(0)}$  which would contradict that  $j \in \mathcal{B}^{(\ell)}$ . Then, since every client in  $(\alpha^{(0)}, z^{(0)})$  is decided (Invariant 4),  $j$  must have been witnessed by some facility  $i$  in  $(\alpha^{(0)}, z^{(0)})$ . Since  $j$  is undecided, by Corollary 8.6 we may further assume  $i \neq i^+$  and thus  $z_i^{(\ell)} = z_i^{(0)}$ . Now suppose toward contradiction that  $\alpha_j^{(0)} > 20^4\theta_s$ . Lemma 8.8 then implies  $\alpha_{j'}^{(0)} \geq 20^{-2}\alpha_j^{(0)} > 20^2\theta_s$  for every  $j' \in N^{(0)}(i)$ . Then, by Lemma 8.9,  $\alpha_{j'} = \alpha_{j'}^{(0)}$  for all  $j' \in N^{(0)}(i)$ . Thus, as  $z_i = z_i^{(0)}$ ,  $i$  is still tight and a witness for  $i$ , which contradicts the assumption that  $j \in \mathcal{B}^{(\ell)}$ , since  $j$  is then decided.

**Case 3:  $j \in \mathcal{B}^{(\ell)}$  was removed by Rule 5.** Let  $j_1, j_2, \dots, j_p$  be the clients in  $\mathcal{B}^{(\ell)}$  that were removed from  $A$  by Rule 5 in the  $\ell$ th call to SWEEP. We index these clients according to the order in which they were removed from  $A$ . The previous cases already imply that the clients in  $\mathcal{B}^{(\ell)} \setminus \{j_1, \dots, j_p\}$  satisfy the induction hypothesis. We now assume that it is true for the clients in  $\mathcal{B}_{<a}^{(\ell)} := \mathcal{B}^{(\ell)} \setminus \{j_a, \dots, j_p\}$  and show that it is also true for client  $j_a$ , i.e., for all clients in  $\mathcal{B}^{(\ell)} \setminus \{j_{a+1}, \dots, j_p\}$ . Consider client  $j_a$ . Then, we must have  $\bar{\alpha}_{j_a}^{(\ell)} \geq d(j_a, j') + \bar{\alpha}_{j'}^{(\ell)}$  for some  $j'$  that was previously removed from  $A$ . Moreover, since  $j_a$  is undecided, Property 2 implies that  $j'$  must also be undecided. We now show that  $j' \in \mathcal{B}^{(\ell)}$ . Indeed, since  $j'$  is undecided, if  $j' \notin \mathcal{B}^{(\ell)}$ , there must be a  $j''$  such that  $2\bar{\alpha}_{j'}^{(\ell)} \geq d(j', j'') + 6\bar{\alpha}_{j''}^{(0)}$ . But then

$$2\bar{\alpha}_{j_a}^{(\ell)} \geq 2d(j_a, j') + 2\bar{\alpha}_{j'}^{(\ell)} \geq 2d(j_a, j') + d(j', j'') + 6\bar{\alpha}_{j''}^{(0)} \geq d(j_a, j'') + 6\bar{\alpha}_{j''}^{(0)},$$

which contradicts the fact that  $j_a \in \mathcal{B}^{(\ell)}$ . Now, since  $j' \in \mathcal{B}^{(\ell)}$  was removed from  $A$  before  $j_a$ , we in fact have  $j' \in \mathcal{B}_{<a}^{(\ell)}$ , and so by assumption there exists some  $j''$  and  $s$  such that  $\frac{1}{9}\theta_s \leq \alpha_{j''}^{(0)} \leq 20^4\theta_s$  and  $\bar{\alpha}_{j'}^{(\ell)} \geq d(j', j'') + \bar{\alpha}_{j''}^{(0)}$ . Thus

$$\bar{\alpha}_{j_a}^{(\ell)} \geq d(j_a, j') + \bar{\alpha}_{j'}^{(\ell)} \geq d(j_a, j') + d(j', j'') + \bar{\alpha}_{j''}^{(0)} \geq d(j_a, j'') + \bar{\alpha}_{j''}^{(0)},$$

and so the induction hypothesis holds for  $j_a$  as well (using  $j''$  and  $s$ ).

□

The above lemma says that any client  $j$  that becomes potentially bad in any solution produced by RAISEPRICE (i.e., in  $j \in \mathcal{B}$  for one produced solution) must have  $\alpha_j^{(0)}$  close to a threshold. Our selection of the shift-parameter  $\sigma$  then ensures that this can only happen for a tiny fraction of the clients. This allows us to bound the connection cost of clients in  $\mathcal{B}$  by an arbitrarily small constant fraction (depending on the parameter  $K$ ) of  $\sum_{j \in \mathcal{D}} \alpha_j$ . However, as stated in the second inequality of Property 2 in Definition 5.1, we need to bound the total connection cost of these clients as a tiny fraction of  $\text{OPT}_k$  instead of  $\sum_{j \in \mathcal{D}} \alpha_j$ . As all we know is that  $\text{OPT}_k \geq \sum_{j \in \mathcal{D}} \alpha_j - \lambda k$ , this requires further arguments that we present in the next section.

We complete this section by formally showing that a client is unlikely to become potentially bad over the randomness of the shift-parameter  $\sigma$ . For any given integer  $\sigma \in [0, K/2)$ , let

$$\mathcal{W}(\sigma) = \{j \in \mathcal{D} : 81^{-1} \cdot 20^{-2} \cdot \theta_s \leq \alpha_j^{(0)} \leq 25 \cdot 20^6 \cdot \theta_s \text{ for some } \theta_s\}.$$

Note that by the above lemma, any client that is in  $\mathcal{B}$  in any solution produced during the considered call to RAISEPRICE, is in  $\mathcal{W}(\sigma)$ .<sup>8</sup> Note that each value  $\alpha_j^{(0)}$  is fixed at the beginning of RAISEPRICE, and there are only a (relatively) small number of choices for  $\sigma$  such that any given  $j$  is in  $\mathcal{W}(\sigma)$ . Thus, if we choose  $\sigma$  uniformly at random, the probability that any given  $j \in \mathcal{W}(\sigma)$  is small. The following corollary formalizes this intuition.

**Corollary 8.12.** *If we select the shift-parameter  $\sigma$  uniformly at random from  $[0, K/2)$ ,*

$$\Pr[j \in \mathcal{W}(\sigma)] \leq \gamma^4 \quad \text{for any client } j.$$

*Proof.* Suppose that we select an integer  $\sigma$  uniformly at random from  $[0, K/2)$ . Then, note that by definition  $\theta_s = (\max_{j \in U^{(0)}} \alpha_j + 2\epsilon_z)(1 + \epsilon)^{K \cdot (s-1) + \sigma}$  and so  $j \in \mathcal{W}(\sigma)$  if and only if:

$$K_1 + K(s-1) + \sigma \leq \log_{1+\epsilon} \alpha_j^{(0)} \leq K_2 + K(s-1) + \sigma,$$

for some  $s$ , where  $K_1 = \log_{1+\epsilon}(81^{-1} \cdot 20^{-2}) + \log_{1+\epsilon}(\max_{j \in U^{(0)}} \alpha_j^{(0)} + 2\epsilon_z)$  and  $K_2 = \log_{1+\epsilon}(25 \cdot 20^6) + \log_{1+\epsilon}(\max_{j \in U^{(0)}} \alpha_j^{(0)} + 2\epsilon_z)$ . In other words,  $\sigma$  needs to satisfy

$$K_1 + K(s-1) - \log_{1+\epsilon} \alpha_j^{(0)} \leq -\sigma \leq K_2 + K(s-1) - \log_{1+\epsilon} \alpha_j^{(0)} \text{ for some integer } s.$$

Notice that the difference between the upper bound and the lower bound is  $K_2 - K_1 = \log_{1+\epsilon}(81 \cdot 25 \cdot 20^8)$  which by selection of  $K$  and  $C_0$  is at most  $\frac{\gamma^4}{2}K$ . Moreover, as  $\sigma \in [0, K/2)$  there is at most one value of  $s$  that can satisfy the above inequalities. It follows that there are at most  $\frac{\gamma^4}{2}K$  distinct values of  $\sigma$  so that  $j \in \mathcal{W}(\sigma)$ . Thus,  $j \in \mathcal{W}(\sigma)$  with probability at most  $\gamma^4$ .  $\square$

## 8.5 Handling dense clients

Corollary 8.12 implies that by carefully selecting our thresholds, we can ensure that only an arbitrarily small fraction  $\gamma^4$  of clients  $j$  can ever appear in  $\mathcal{B}$  throughout the execution of RAISEPRICE. As briefly discussed previously, this is unfortunately insufficient for our purposes. Specifically, in order to charge the extra service cost incurred by this small fraction of clients to  $\text{OPT}_k \geq \sum_{j \in \mathcal{D}} \alpha_j - \lambda k$ , we need to handle carefully those clients  $j$  for which most of  $\alpha_j$  is contributing toward the opening cost  $\lambda k$ .

To cope with this difficulty, we introduce the notion *dense* facilities and clients, as follows. Recall that  $\gamma \ll \epsilon$  is a small constant. We define the  $\gamma$ -close neighborhood of a facility  $i$  as

$$N_\gamma^{(0)}(i) = \{j \in \mathcal{D} : d(j, i)^2 \leq \gamma \alpha_j^{(0)}\}.$$

Then, we say that a facility  $i \in \mathcal{IS}^{(0)}$  is *dense* if

$$\sum_{j \in N_\gamma^{(0)}(i)} \alpha_j^{(0)} \geq (1 - \gamma) z_i^{(0)}.$$

We let  $\mathcal{F}_\mathcal{D} \subseteq \mathcal{IS}^{(0)}$  be the set of all dense facilities, and then define the set of *dense clients* as  $\mathcal{D}_\mathcal{D} = \bigcup_{i \in \mathcal{F}_\mathcal{D}} N_\gamma^{(0)}(i)$ . Note that the  $\gamma$ -close neighborhoods, dense facilities, and dense clients

---

<sup>8</sup>To argue  $\mathcal{B} \subseteq \mathcal{W}(\sigma)$ , the bounds  $81^{-1} \theta_s \leq \alpha_j^{(0)} \leq 25 \cdot 20^4 \cdot \theta_s$  for some  $\theta_s$  would be sufficient in the definition of  $\mathcal{W}(\sigma)$ . However, the more relaxed bounds will be useful when analyzing dense clients in the next section.

are all determined only by the input solution  $(\alpha^{(0)}, z^{(0)})$  and the integral solution  $\text{IS}^{(0)}$  passed to RAISEPRICE.

Intuitively, the dense clients are precisely those troublesome clients for which  $\alpha_j^{(0)}$  is much larger than the service cost of  $j$  in  $\text{IS}^{(0)}$ . In order to avoid paying  $36\alpha_j^{(0)}$  for any such clients, we construct a set of *special facilities*  $\mathcal{F}_s^{(\ell)}$  and *special clients*  $\mathcal{D}_s^{(\ell)}$  for each  $\alpha^{(\ell)}$  produced by our RAISEPRICE, as follows. Let

$$\begin{aligned} \mathcal{F}_s^{(\ell)} &= \{i \in \mathcal{F}_D : |N_\gamma^{(0)}(i) \cap \mathcal{B}| \neq \emptyset \text{ and } \alpha_j^{(\ell)} \geq \alpha_j^{(0)} \forall j \in N_\gamma^{(0)}(i)\}, \quad \text{and} \\ \mathcal{D}_s^{(\ell)}(i) &= N_\gamma^{(0)}(i) \quad \text{for every } i \in \mathcal{F}_s^{(\ell)}, \end{aligned} \tag{8.5}$$

We will show that each solution  $\mathcal{S}^{(\ell)} = (\alpha^{(\ell)}, z^{(\ell)}, \mathcal{F}_s^{(\ell)}, \mathcal{D}_s^{(\ell)})$  is roundable, with the set of remaining bad clients given by  $\mathcal{D}_B = \mathcal{B} \setminus \mathcal{D}_D$ .

Recall that in Definition 5.1, we have  $\tau_i = \max_{j \in N(i) \cap \mathcal{D}_S(i)} \alpha_j$  for any facility  $i \in \mathcal{F}_S$  and  $\tau_i = t_i$  for all other facilities. Note that as  $(\alpha^{(0)}, z^{(0)})$  by Invariant 4 is a completely decided solution, by our choice of  $\mathcal{F}_S$  and  $\mathcal{D}_S$ , we have  $\mathcal{F}_s^{(0)} = \emptyset$  in the roundable solution  $(\alpha^{(0)}, z^{(0)}, \mathcal{F}_s^{(0)}, \mathcal{D}_s^{(0)})$ . Therefore, the conflict graph  $H^{(0)}$  of  $(\alpha^{(0)}, z^{(0)}, \mathcal{F}_s^{(0)}, \mathcal{D}_s^{(0)})$  does not contain any special facilities, and so  $\tau_i = t_i$  for each facility  $i \in H^{(0)}$ . Moreover, recall that  $\text{IS}^{(0)}$  was the maximal independent set of  $H^{(0)}$  computed in the previous call to GRAPHUPDATE. In particular,  $|\text{IS}^{(0)}| > k$  and  $\text{IS}^{(0)}$  does not contain any special facilities.

The following simple lemma is now a direct consequence of our definitions.

**Lemma 8.13.** *Suppose that  $j \in N_\gamma^{(0)}(i)$  for some  $i \in \text{IS}^{(0)}$ . Then,  $\beta_{i'j}^{(0)} = 0$  for all other  $i' \in \text{IS}^{(0)}$ . Moreover, for every client  $j \in \mathcal{D}$ ,  $\alpha_j^{(0)} \geq \sum_{i \in \text{IS}^{(0)}} \beta_{ij}^{(0)}$ .*

*Proof.* We start by proving that if  $j \in N_\gamma^{(0)}(i)$  for some  $i \in \text{IS}^{(0)}$ , then  $\beta_{i'j}^{(0)} = 0$  for all other  $i' \in \text{IS}^{(0)}$ . Consider some facility  $i \in \text{IS}^{(0)}$ , and suppose that  $j \in N_\gamma^{(0)}(i)$ . Further, suppose that for some other facility  $i' \in H^{(0)}$  we have  $\beta_{i'j} > 0$ . Note that this implies (since no facility is special) that  $j$  is adjacent to both  $i$  and  $i'$  in the client-facility graph that generated  $H^{(0)}$ . We shall show that  $i' \notin \text{IS}^{(0)}$ . Indeed, we must have:

$$d(i, i') \leq d(i, j) + d(j, i') < \sqrt{\gamma} \cdot \bar{\alpha}_j^{(0)} + \bar{\alpha}_j^{(0)} < \sqrt{\delta} \cdot \bar{\alpha}_j^{(0)} \leq \sqrt{\delta t_i} = \sqrt{\delta \tau_i}.$$

Thus, there is an edge between  $i$  and  $i'$  in the conflict graph  $H^{(0)}$ , and since  $i \in \text{IS}^{(0)}$ , we have  $i' \notin \text{IS}^{(0)}$ .

We shall now prove that  $\alpha_j^{(0)} \geq \sum_{i \in \text{IS}^{(0)}} \beta_{ij}^{(0)}$  for any client  $j \in \mathcal{D}$ . Again using that no facility is special, we have that  $j$ 's neighborhood in the client-facility graph that generated  $H^{(0)}$  is equal to the set of tight facilities that  $j$  is paying for. Therefore, we have that  $\alpha_j^{(0)} - \sum_{i \in \text{IS}^{(0)}} \beta_{ij}^{(0)} \geq d(j, \text{IS}^{(0)})^2 / \rho$  (which implies  $\alpha_j^{(0)} \geq \sum_{i \in \text{IS}^{(0)}} \beta_{ij}^{(0)}$ ) by the exact same arguments as “Case  $s = 1$ ” and “Case  $s > 1$ ” in the proof of Theorem 3.4.  $\square$

The next lemma formalizes our intuition, showing that we can indeed charge the total  $\alpha^{(0)}$ -value of all non-dense clients to  $\text{OPT}_k$ .

**Lemma 8.14.**  $\sum_{j \in \mathcal{D} \setminus \mathcal{D}_D} \alpha_j^{(0)} \leq \gamma^{-3} \cdot \text{OPT}_k.$



*Proof.* We partition the clients in  $\mathcal{D} \setminus \mathcal{D}_D$  into two sets:

$$\mathcal{D}_{>\gamma} = \{j \in \mathcal{D} \setminus \mathcal{D}_D : d(j, \text{IS}^{(0)})^2 > \gamma \cdot \alpha_j^{(0)}\} \text{ and } \mathcal{D}_{\leq\gamma} = \{j \in \mathcal{D} \setminus \mathcal{D}_D : d(j, \text{IS}^{(0)})^2 \leq \gamma \cdot \alpha_j^{(0)}\}.$$

By definition,

$$\sum_{j \in \mathcal{D}_{>\gamma}} d(j, \text{IS}^{(0)})^2 > \gamma \cdot \sum_{j \in \mathcal{D}_{>\gamma}} \alpha_j^{(0)}. \quad (8.6)$$

To bound the remaining clients consider the following fractional token argument: each client  $j \in \mathcal{D}_{>\gamma}$  distributes  $\beta_{ij}^{(0)} = [\alpha_j^{(0)} - d(j, i)^2]^+$  tokens to each facility  $i \in \text{IS}^{(0)}$ . Lemma 8.13 says that  $\sum_{i \in \text{IS}^{(0)}} \beta_{ij}^{(0)} \leq \alpha_j^{(0)}$  for every client  $j$ , and so the total number of tokens distributed is at most  $\sum_{j \in \mathcal{D}_{>\gamma}} \alpha_j^{(0)}$ .

Now, we observe that for each client  $j \in \mathcal{D}_{\leq\gamma}$ , the closest facility  $i \in \text{IS}^{(0)}$  to  $j$  must *not* be in  $\mathcal{F}_D$ , since otherwise  $j$  would be in  $\mathcal{D}_D$ . Hence, we have

$$\sum_{j \in N_\gamma^{(0)}(i)} (1 - \gamma) \alpha_j^{(0)} \leq \sum_{j \in N_\gamma^{(0)}(i)} [\alpha_j^{(0)} - d(j, i)^2]^+ \leq (1 - \gamma) z_i^{(0)}. \quad (8.7)$$

Moreover, there must be at least  $\gamma z_i^{(0)}$  tokens assigned to  $i$ , because it is a tight facility with respect to  $\alpha^{(0)}$  and by Lemma 8.13, every client  $j \notin \mathcal{D}_{>\gamma} \cup N_\gamma^{(0)}(i)$  must have  $\beta_{ij} = 0$ . Therefore,

$$\sum_{j \in \mathcal{D}_{>\gamma}} \beta_{ij}^{(0)} = \sum_{j \in \mathcal{D} \setminus N_\gamma^{(0)}(i)} \beta_{ij}^{(0)} \geq \gamma z_i^{(0)}.$$

Thus,

$$\sum_{j \in \mathcal{D}_{\leq\gamma}} \alpha_j^{(0)} = \sum_{i \in \text{IS}^{(0)} \setminus \mathcal{F}_D} \sum_{j \in N_\gamma^{(0)}(i)} \alpha_j^{(0)} \leq \sum_{i \in \text{IS}^{(0)} \setminus \mathcal{F}_D} z_i^{(0)} = \frac{1}{\gamma} \sum_{i \in \text{IS}^{(0)} \setminus \mathcal{F}_D} \gamma \cdot z_i^{(0)} \leq \frac{1}{\gamma} \sum_{j \in \mathcal{D}_{>\gamma}} \alpha_j^{(0)},$$

where the first equality follows from Lemma 8.13, the first inequality from (8.7), and the last inequality from the fact that each facility  $i \in \text{IS}^{(0)} \setminus \mathcal{F}_D$  received at least  $\gamma z_i^{(0)}$  tokens and the total amount of distributed tokens was at most  $\sum_{j \in \mathcal{D}_{>\gamma}} \alpha_j^{(0)}$ .

Hence,

$$\begin{aligned} \sum_{j \in \mathcal{D}_{>\gamma}} \alpha_j^{(0)} + \sum_{j \in \mathcal{D}_{\leq\gamma}} \alpha_j^{(0)} &\leq \left(1 + \frac{1}{\gamma}\right) \sum_{j \in \mathcal{D}_{>\gamma}} \alpha_j^{(0)} \\ &< \frac{1}{\gamma} \cdot \left(1 + \frac{1}{\gamma}\right) \sum_{j \in \mathcal{D}_{>\gamma}} d(j, \text{IS}^{(0)})^2 \\ &\leq \frac{1}{\gamma} \cdot \left(1 + \frac{1}{\gamma}\right) (\rho + 1000\epsilon) \text{OPT}_k, \end{aligned}$$

where the penultimate inequality follows from (8.6) and the last inequality from Theorem 6.4.  $\square$

Lemma 8.14 shows that we can relate the total  $\alpha$ -value of all non-dense clients to  $\text{OPT}_k$ , as desired. Moreover, we have the following corollary.

**Corollary 8.15.** *If we select the shift-parameter  $\sigma$  uniformly at random from  $[0, K/2)$ ,*

$$\mathbb{E} \left[ \sum_{j \in \mathcal{W}(\sigma) \setminus \mathcal{D}_D} \alpha_j^{(0)} \right] \leq \gamma \cdot \text{OPT}_k.$$

*In particular, if we set  $\mathcal{W} = \mathcal{W}(\sigma)$  for the value  $\sigma$  that minimizes  $\sum_{j \in \mathcal{W}(\sigma) \setminus \mathcal{D}_D} \alpha_j^{(0)}$  then, we have*

$$\sum_{j \in \mathcal{W} \setminus \mathcal{D}_D} \alpha_j^{(0)} \leq \gamma \cdot \text{OPT}_k.$$

*Proof.* For the first claim, we note that

$$\mathbb{E} \left[ \sum_{j \in \mathcal{W}(\sigma) \setminus \mathcal{D}_D} \alpha_j^{(0)} \right] = \sum_{j \in \mathcal{D} \setminus \mathcal{D}_D} \Pr[j \in \mathcal{W}(\sigma)] \cdot \alpha_j^{(0)} \leq \gamma^4 \sum_{j \in \mathcal{D} \setminus \mathcal{D}_D} \alpha_j^{(0)} \leq \gamma \cdot \text{OPT}_k,$$

where the first inequality follows from by Corollary 8.12 and the last inequality follows from Lemma 8.14. The second claim now follows since the minimum of left-hand side over all  $\sigma \in [0, K/2)$  is at most its expected value over a randomly chosen  $\sigma \in [0, K/2)$ .  $\square$

**Remark 8.16.** *The only property of the selection of  $\sigma$  that we use is that  $\sum_{j \in \mathcal{W} \setminus \mathcal{D}_D} \alpha_j^{(0)} \leq \gamma \cdot \text{OPT}_k$ . It easy to find the  $\sigma$  that minimizes  $\sum_{j \in \mathcal{W}(\sigma) \setminus \mathcal{D}_D} \alpha_j^{(0)}$  (since the number of options is constant) at the start of a call to RAISEPRICE and thus the selection of the shift-parameter can be derandomized.*

Now, we show how to obtain a better bound than that given by Lemma 8.7 for dense clients  $\mathcal{D}_D \cap \mathcal{B}$ . This will allow us to eventually obtain a roundable solution. For this purpose, recall the definition of *special* facilities and clients  $\mathcal{F}_S$  and  $\mathcal{D}_S$ :

$$\begin{aligned} \mathcal{F}_S &= \{i \in \mathcal{F}_D : |N_\gamma^{(0)}(i) \cap \mathcal{B}| \neq \emptyset \text{ and } \alpha_j \geq \alpha_j^{(0)} \ \forall j \in N_\gamma^{(0)}(i)\} \quad \text{and} \\ \mathcal{D}_S(i) &= N_\gamma^{(0)}(i) \quad \text{for every } i \in \mathcal{F}_S. \end{aligned}$$

Also, recall that for each special facility  $i \in \mathcal{F}_S$  we define  $\tau_i = \max_{j \in N(i) \cap \mathcal{D}_S(i)} \alpha_j$  and for all other facilities  $i$  we let  $\tau_i = t_i = \max_{j \in N(i)} \alpha_j$ . Note that  $\tau_i \leq t_i$  for all  $i \in \mathcal{F}$ . We now show how to bound the cost of all clients in  $\mathcal{D}_D \cap \mathcal{B}$  using the facilities of  $\mathcal{F}_S$ .

**Lemma 8.17.** *For any  $j \in \mathcal{D}_D \cap \mathcal{B}$ , either:*

- *There exists a tight facility  $i \in \mathcal{F}$  such that  $(1 + \sqrt{\delta} + 10\epsilon)\bar{\alpha}_j \geq d(j, i) + \sqrt{\delta t_i}$ .*
- *There exists a special facility  $i \in \mathcal{F}_S$  such that  $(1 + \sqrt{\delta} + 10\epsilon)\bar{\alpha}_j \geq d(j, i) + \sqrt{\delta \tau_i}$ .*

*Proof.* Consider a client  $j_0 \in \mathcal{D}_D \cap \mathcal{B}$ . Since  $j_0 \in \mathcal{D}_D$  there must be some  $i^* \in \mathcal{F}_D$  such that  $j_0 \in N_\gamma^{(0)}(i^*)$ . Moreover, since  $j_0 \in \mathcal{B}$ ,  $j_0$  is undecided and so by Lemma 8.4 we must have  $\alpha_{j_0} \geq \alpha_{j_0}^{(0)}$ .

Suppose first that  $i^* \in \mathcal{F}_S$ . Then  $\tau_{i^*} = \max_{j \in N(i^*) \cap \mathcal{D}_S(i^*)} \alpha_j$ . Since  $i^* \in \mathcal{F}_S$  we have  $\alpha_j \geq \alpha_j^{(0)}$  for all  $j \in N_\gamma^{(0)}(i^*)$ . We claim that  $\tau_{i^*} \leq (1 + \epsilon)^2 \alpha_{j_0}$ . Indeed, otherwise there is a client  $j \in$

$N(i^*) \cap \mathcal{D}_s(i^*) = N(i^*) \cap N_\gamma^{(0)}(i^*)$  such that  $\bar{\alpha}_j > (1 + \epsilon)\bar{\alpha}_{j_0}$  and so

$$\begin{aligned}
(1 + \epsilon)\bar{\alpha}_j &> (1 + \epsilon)\bar{\alpha}_{j_0} + \frac{\epsilon}{2} \cdot \bar{\alpha}_{j_0} + \frac{\epsilon}{2} \cdot \bar{\alpha}_j \\
&\geq (1 + \epsilon)\bar{\alpha}_{j_0} + \frac{\epsilon}{2} \cdot \bar{\alpha}_{j_0}^{(0)} + \frac{\epsilon}{2} \cdot \bar{\alpha}_j^{(0)} && (\alpha_{j_0} \geq \alpha_{j_0}^{(0)} \text{ and } \alpha_j \geq \alpha_j^{(0)}) \\
&\geq (1 + \epsilon)\bar{\alpha}_{j_0} + \frac{\epsilon}{2\sqrt{\gamma}} \cdot d(j_0, i^*) + \frac{\epsilon}{2\sqrt{\gamma}} \cdot d(j, i^*) && (j, j_0 \in N_\gamma^{(0)}(i^*)) \\
&\geq (1 + \epsilon)\bar{\alpha}_{j_0} + (1 + \epsilon)d(j_0, i^*) + (1 + \epsilon)d(j, i^*) && (\gamma \ll \epsilon \text{ and so } \frac{\epsilon}{2\sqrt{\gamma}} \geq (1 + \epsilon)) \\
&\geq (1 + \epsilon)(\bar{\alpha}_{j_0} + d(j_0, j)),
\end{aligned}$$

contradicting Invariant 3, since the  $\alpha$ -ball of  $j$  would then strictly contain the  $\alpha$ -ball of  $j_0$ . Hence,  $\sqrt{\tau_{i^*}} \leq (1 + \epsilon)\bar{\alpha}_{j_0}$ . Furthermore,  $d(j_0, i^*) \leq \gamma\bar{\alpha}_{j_0}^{(0)} \leq \gamma\bar{\alpha}_{j_0}$  and therefore

$$(1 + \sqrt{\delta} + 3\epsilon)\bar{\alpha}_{j_0} \geq d(j_0, i^*) + \sqrt{\delta\tau_{i^*}}.$$

On the other hand, if  $i^* \notin \mathcal{F}_s$  then (by the definition of  $\mathcal{F}_s$ ) there must be some  $j \in N_\gamma^{(0)}(i^*)$  with  $\alpha_j < \alpha_j^{(0)}$ . By Lemma 8.4,  $j$  must be decided. Then,  $j \notin \mathcal{B}$  and so by Lemma 8.7 there exists some tight facility  $i$  such that  $(1 + \sqrt{\delta} + \epsilon)\bar{\alpha}_j \geq d(j, i) + \sqrt{\delta t_i}$ . Moreover, applying the same argument as above, we must have  $\bar{\alpha}_j^{(0)} \leq (1 + \epsilon)\bar{\alpha}_{j_0}^{(0)}$ , since otherwise in  $\alpha^{(0)}$ , the  $\alpha$ -ball of  $j$  would strictly contain the  $\alpha$ -ball of  $j_0$ , contradicting Invariant 3. Then, we have:

$$\begin{aligned}
d(j_0, i) + \sqrt{\delta t_i} &\leq d(j_0, i^*) + d(i^*, j) + d(j, i) + \sqrt{\delta t_i} \\
&\leq \gamma\bar{\alpha}_{j_0}^{(0)} + \gamma\bar{\alpha}_j^{(0)} + (1 + \sqrt{\delta} + \epsilon)\bar{\alpha}_j \\
&\leq \epsilon\bar{\alpha}_{j_0}^{(0)} + \epsilon(1 + \epsilon)\bar{\alpha}_{j_0}^{(0)} + (1 + \sqrt{\delta} + \epsilon)(1 + \epsilon)\bar{\alpha}_{j_0}^{(0)} \\
&< (1 + \sqrt{\delta} + 10\epsilon)\bar{\alpha}_{j_0},
\end{aligned}$$

where for the final inequality we used that  $j_0$  is undecided and so by Lemma 8.4 we have  $\alpha_{j_0} \geq \alpha_{j_0}^{(0)}$ .  $\square$

## 8.6 Showing that each solution is roundable and completing the analysis

We start by showing that each solution  $(\alpha, z)$  produced by Algorithm 1 satisfies the properties of Definition 5.1.

**Proposition 8.18.** *Every solution  $(\alpha, z)$  produced by Algorithm 1 is roundable.*

*Proof.* By construction, each solution  $(\alpha, z)$  produced by Algorithm 1 is feasible with respect to  $\text{DUAL}(\lambda + \epsilon_z)$  and  $\epsilon_z < \frac{1}{n}$ . In addition, we have  $\lambda \leq z_i \leq \lambda + \epsilon_z \leq \lambda + 1/n$  for all  $i \in \mathcal{F}$ . It remains to show that Properties 2 and 3 of Definition 5.1 are satisfied. Recall the definitions of  $\mathcal{F}_D$  and  $\mathcal{D}_D$ , and define  $\mathcal{F}_s$  and  $\mathcal{D}_s$  as in (8.5). Further let  $\mathcal{D}_B = \mathcal{W} \setminus \mathcal{D}_D$ .

Now we show that Property 2 holds for  $\mathcal{S} = (\alpha, z, \mathcal{F}_s, \mathcal{D}_D)$  with respect to the set  $\mathcal{D}_B$ . By Lemma 8.11 (which shows that a client  $j \in \mathcal{B}$  only if  $\frac{1}{81}\theta_s \leq \alpha_j^{(0)} \leq 25 \cdot 20^4\theta_s$  for some stage  $s$ ) we have  $\mathcal{B} \subseteq \mathcal{W}$ . Thus  $\mathcal{B} \setminus \mathcal{D}_D \subseteq \mathcal{D}_B$ . Now, by Lemma 8.7 for all  $j \in \mathcal{D} \setminus \mathcal{B}$  there exists some tight facility  $i$  such that

$$(1 + \sqrt{\delta} + 10\epsilon)^2 \alpha_j \geq (d(j, i) + \sqrt{\delta t_i})^2 \geq (d(j, i) + \sqrt{\delta \tau_i})^2.$$

Similarly, by Lemma 8.17, for all  $j \in \mathcal{B} \cap \mathcal{D}_D$ , there exists either some tight facility  $i$  or some special facility  $i \in \mathcal{F}_s$  such that

$$(1 + \sqrt{\delta} + 10\epsilon)^2 \alpha_j \geq (d(j, i) + \sqrt{\delta \tau_i})^2.$$

Finally, for each remaining client  $j \in \mathcal{B} \setminus \mathcal{D}_D \subseteq \mathcal{D}_B$ , by Lemma 8.7, there is some tight facility  $i$  such that

$$36\alpha_j^{(0)} \geq \left(d(j, i) + \sqrt{\delta t_i}\right)^2 \geq \left(d(j, i) + \sqrt{\delta \tau_i}\right)^2.$$

For each such client  $j$ , let  $w(j)$  be this specified tight facility  $i$ . Then,

$$\sum_{j \in \mathcal{D}_B} \left(d(j, i) + \sqrt{\delta \tau_{w(j)}}\right)^2 \leq 36 \sum_{j \in \mathcal{D}_B} \alpha_j^{(0)} \leq 36\gamma \cdot \text{OPT}_k,$$

where the last inequality follows from Corollary 8.15.

Finally, we show that Property 3 must hold. Consider some  $i \in \mathcal{F}_S$ . By definition of  $\mathcal{F}_S$ , we must have  $j \in \mathcal{B}$  for some  $j \in N_\gamma^{(0)}(i)$ . Then, by Lemma 8.11 we must have  $\frac{1}{81}\theta_s \leq \alpha_j^{(0)} \leq 25 \cdot 20^4 \theta_s$  for some  $s$ . By Lemma 8.8 (which bounds the ratio to be at most  $19^2 \leq 20^2$  between  $\alpha_j$  and  $\alpha_{j'}$  for any pair of clients  $j, j'$  that share a tight edge to some common facility  $i$ ), we must have  $\alpha_{j'}^{(0)} \in \mathcal{W}$  for any  $j' \in N^{(0)}(i)$ . Moreover, by Lemma 8.13 (which shows that each dense client pays for at most one dense facility in  $\text{IS}^{(0)}$ ), we have  $\beta_{ij}^{(0)} = 0$  for all  $j \in \mathcal{D}_D \setminus N_\gamma^{(0)}(i)$ . Altogether, then we have  $N^{(0)}(i) \subseteq \mathcal{W}$  and  $N^{(0)}(i) \cap \mathcal{D}_D = N_\gamma^{(0)}(i)$  and so

$$N^{(0)}(i) \setminus N_\gamma^{(0)}(i) = N^{(0)}(i) \setminus \mathcal{D}_D \subseteq \mathcal{W} \setminus \mathcal{D}_D,$$

for every  $i \in \mathcal{F}_S$ . Notice that Lemma 8.13 also implies that for each dense facility  $i \in \mathcal{F}_D$  there is some dense client that only pays for that facility in  $\text{IS}^{(0)}$ , so indeed  $|\mathcal{F}_S| \leq |\mathcal{F}_D| \leq n$ . It remains to prove that  $\sum_{i \in \mathcal{F}_S} \sum_{j \in \mathcal{D}_S} \beta_{ij} \geq \lambda |\mathcal{F}_S| - \gamma \cdot \text{OPT}_k$ . To that end, recall that Invariant 4 implies that  $\mathcal{F}_S^{(0)} = \emptyset$ . Thus, every  $i \in \text{IS}^{(0)}$  must have been tight in  $\alpha^{(0)}$ , and hence  $\sum_{j \in N^{(0)}(i)} \beta_{ij}^{(0)} \geq z_i^{(0)} \geq \lambda$ . Combining these observations, for every  $i \in \mathcal{F}_S$  we have:

$$\sum_{j \in \mathcal{D}_S(i)} \beta_{ij} = \sum_{j \in N_\gamma^{(0)}(i)} \beta_{ij} \geq \sum_{j \in N_\gamma^{(0)}(i)} \beta_{ij}^{(0)} \geq \lambda - \sum_{j \in N^{(0)}(i) \setminus N_\gamma^{(0)}(i)} \beta_{ij}^{(0)} \geq \lambda - \sum_{j \in \mathcal{W} \setminus \mathcal{D}_D} \beta_{ij}^{(0)}, \quad (8.8)$$

where the first inequality follows from the definition of  $\mathcal{F}_S$ , which requires that for any  $i \in \mathcal{F}_S$ ,  $\alpha_j \geq \alpha_j^{(0)}$  for all  $j \in N_\gamma^{(0)}(i)$ . By Invariant 4, every client is decided in  $\alpha^{(0)}$  and so, in particular,  $\mathcal{F}_S^{(0)} = \emptyset$  and  $\text{IS}^{(0)}$  contains no special facilities. Then, by Lemma 8.13,  $\sum_{i \in \mathcal{F}_S} \beta_{ij}^{(0)} \leq \sum_{i \in \text{IS}^{(0)}} \beta_{ij}^{(0)} \leq \alpha_j^{(0)}$  for all  $j$ . Summing (8.8) over all  $i \in \mathcal{F}_S$  we thus have

$$\sum_{i \in \mathcal{F}_S} \sum_{j \in \mathcal{D}_S(i)} \beta_{ij} \geq |\mathcal{F}_S| \lambda - \sum_{j \in \mathcal{W} \setminus \mathcal{D}_D} \sum_{i \in \mathcal{F}_S} \beta_{ij}^{(0)} \geq |\mathcal{F}_S| \lambda - \sum_{j \in \mathcal{W} \setminus \mathcal{D}_D} \alpha_j^{(0)} \geq |\mathcal{F}_S| \lambda - \gamma \cdot \text{OPT}_k,$$

where the final inequality follows from Corollary 8.15.  $\square$

The following theorem completes the analysis.

**Theorem 8.19.** *RAISEPRICE runs in polynomial time and produces a polynomial number of close roundable solutions.*

*Proof.* That the produced solutions are close follows from Proposition 8.10 and the produced solutions are roundable follows from Proposition 8.18. We continue to bound the running time and the number of produced solutions. RAISEPRICE produces one solution for each call to SWEEP. In Appendix A we argue (similarly as we did for QUASISWEEP) that SWEEP can be implemented in polynomial time, and it is clear that the remaining operations in RAISEPRICE can be implemented

in polynomial time. Thus, to prove both claims, it suffices bound the number of calls to SWEEP in RAISEPRICE. For that purpose, define:

$$M = \lambda + \max_{j \in \mathcal{D}, i \in \mathcal{F}} d(j, i) \leq 4n^7 + n^6 < n^8.$$

Using our preprocessing (Lemma 4.1), we note that at all times during any call to RAISEPRICE, we have  $\alpha_j \leq M$ , since otherwise  $\alpha$  would be infeasible (contradicting Invariant 2).

Let us now bound the number of calls to SWEEP in each stage. In stage 1, we make only 1 call to SWEEP, as shown in Lemma 8.3. In each stage  $s > 1$ , RAISEPRICE calls SWEEP only until  $\alpha_j \geq \theta_s$  and  $\alpha_j \geq \alpha_j^{(0)}$  for every undecided client  $j$ . Consider a call to SWEEP in stage  $s > 1$  and let  $(\alpha, z)$  be the produced solution. Let  $j$  be the undecided client with the smallest  $\alpha$ -value in  $(\alpha, z)$  (breaking ties in the order of removal from the set  $A$ ). If  $j$  was removed by Rule 4, we have  $\alpha_j \geq \theta_s$  and  $\alpha_j \geq \alpha_j^{(0)}$  and so every undecided client has an  $\alpha$ -value of at least  $\theta_s$  which implies the termination of stage  $s$ . Otherwise, as  $j$  has the smallest  $\alpha$ -value of undecided clients it cannot be removed by Rule 5 (by Property 2) and so it must have been removed by Rule 3. Therefore, by the definition of that rule,  $j$  was undecided in the previous iteration and its  $\alpha$ -value has increased by  $\epsilon_z$  in the considered call to SWEEP. By the above, we have that either stage  $s$  terminates or the smallest  $\alpha$ -value of the undecided clients increases by at least  $\epsilon_z$ . Therefore, the stage must terminate after at most  $\epsilon_z^{-1}M = n^{O(\epsilon^{-1}\gamma^{-4})}$  calls to SWEEP since no  $\alpha$ -value is larger than  $M$ .

Finally, let us bound the number of stages executed in RAISEPRICE. By Lemma 8.5 after stage  $s$ , all clients with  $\alpha_j < \theta_s$  are decided. Then, for  $s = (K\epsilon)^{-1}\Theta(\log n) = \gamma^4 O(\log n)$  we have  $\theta_s > M$  and so all clients must be decided.  $\square$

## A Running time Analysis of SWEEP

In this section, we present a polynomial time implementation of the SWEEP procedure. The algorithm is exactly the same as QUASISWEEP, besides the set of events. Recall that the polynomial time algorithm for QUASISWEEP is as follows: repeatedly find the next event that happens, then update the  $\alpha$ -values. We increase  $\theta$  at a rate 1, so that  $\theta$  corresponds naturally to our notion of time. Let  $\theta^{(0)}$  denote the value of  $\theta$  at the time that the last event has happened.

We now focus on the events, explain them in detail, show the number of times that each can occur, and discuss the way that we can find each one of them. We consider (1) the tight edges, (2) the set  $A$ , (3) the set of potentially tight facilities, denoted by  $P$ , (4) the set of clients with maximum  $\alpha$ -value in  $N(i)$  for each facility in  $i \in P$ , and (5) the buckets of clients and  $\theta$ . While all the above quantities remain the same, the rate at which any client's  $\alpha$ -value is changing remains constant. We now consider the set of events that may cause these quantities to change. We show that we can easily compute the next time that each event would occur if no other event happened before it, and that the number of such events must be polynomial in  $n = |\mathcal{D}|$  and  $m = |\mathcal{F}|$ . Then, the minimum such time is the next time that the behavior of SWEEP might change. We compute the minimum such time after  $\theta^{(0)}$ , then update all of the  $\alpha$ -values (using their current rates of change and the time of this event) to obtain the  $\alpha$ -values at this time. Having these values, we then recompute the quantities (1)-(5) in the following order:

- Update the  $\alpha$ -values: For each  $j \in \mathcal{D}$ , we compute the new value of  $\alpha_j$  by multiplying the previous rate of change in  $\alpha_j$  by  $\theta - \theta^{(0)}$  and adding this to the previous value  $\alpha_j$ .
- The buckets of the  $\alpha$ -values: For each  $\alpha_j$ , we update the current bucket according to the bucketing described in SWEEP. In the case that some  $\alpha_j$  was decreasing and now  $\alpha_j$  is on the lower border of the current bucket, we place  $\alpha_j$  in the next lower bucket.

- Update  $A$ : Now, for each  $j \notin A$ , if  $\alpha_j = \theta$  we add  $j$  to  $A$ . Then, for each  $j \in A$ , we remove  $j$  if one of the conditions of Rules 1-5 in SWEEP is satisfied by the newly updated  $\alpha$ -values. Notice that a client  $j$  might be added and then immediately removed in this step.
- Update the set of tight edges: For each  $j \in \mathcal{D}$  and  $i \in \mathcal{F}$ , if  $\alpha_j > d(j, i)^2$  then add  $j$  to  $N(i)$ . If  $\alpha_j \leq d(j, i)^2$  then we remove  $j$  from  $N(i)$  unless  $j \in A$ . If  $\alpha_j = d(j, i)^2$  and  $j \in A$ , then we add  $j$  to  $N(i)$ .
- Update  $P$ : For each  $i \in \mathcal{F}$ , we compute whether  $i$  is potentially tight as follows. If  $\alpha_j > \alpha_j^{(0)}$  for any  $j \in N(i)$ , then place  $i \in P$ . If  $\alpha_j = \alpha_j^{(0)}$  for some  $j \in N(i) \cap A$  then we also place  $i \in P$ , since as soon as  $\alpha_j$  increases by an infinitesimal amount, we will have  $\alpha_j > \alpha_j^{(0)}$ . No other facilities are placed in  $P$ .

Notice that the above cases capture all the potentially tight facilities as described in the analysis of SWEEP except for the following. We could have some facility  $i$  with  $\alpha_j \leq \alpha_j^{(0)}$  for all  $j \in N(i)$  and  $\alpha_j = \alpha_j^{(0)}$  for all  $j \in N^{(0)}(i)$  but there is no  $j \in N(i) \cap A$  with  $\alpha_j = \alpha_j^{(0)}$ . Then, our definition of SWEEP would consider  $i$  potentially tight, but here we do not. However, observe that in this case  $N(i) \subseteq N^{(0)}(i)$  and so we must have that  $N(i) \cap A = \emptyset$ . Therefore,  $i$  cannot cause any  $\alpha$ -value to decrease in SWEEP, so there is no harm in having  $i \notin P$ .

- Update rates: For each  $j \in A$ , we set the rate of change for  $\alpha_j$  to be 1, and for each  $j$  with  $\alpha_j = t_i$  for some  $i \in P$ , we set the rate of change of  $\alpha_j$  to be  $-|A|$  if and only if  $N(i) \cap A \neq \emptyset$ . For all other clients  $j$ , we set the rate of change for  $\alpha_j$  to 0.

Now, we focus on the events that may require updating the above values. Throughout our discussion we use the fact that once a client's  $\alpha$ -value has been increased by SWEEP, it is not subsequently decreased.

First we focus on (1), i.e., on the tight edges. The following two events take into account when an edge becomes tight and when an edge becomes untight.

- Event 1: The edge between client  $j$  and facility  $i$  becomes tight. Since a client's  $\alpha$ -value is never decreased after it has been increased, this event can happen at most once for each client-facility pair. Notice that this event can occur only when the edge  $(i, j)$  is not tight and  $j \in A$ , so  $\alpha_j$  is increasing. Then, since  $j \in A$ , this event happens when  $d(i, j)^2 = \alpha_j = \theta$ .
- Event 2: The edge between client  $j$  and facility  $i$  becomes untight. Similar to the previous event, this can also happen at most once for each  $j$  and  $i$ . Notice that this event can occur only when the edge  $(i, j)$  is tight and the  $\alpha_j$  value is decreasing. Then,  $\alpha_j$  is decreasing at a rate of  $|A|$ , and so the event happens at time  $\theta$  satisfying  $|A|(\theta - \theta^{(0)}) = \alpha_j - d(i, j)^2$ .

Second, we focus on (2), i.e., the changes that can happen to set  $A$ . Notice that,  $A$  changes only if a client joins it or leaves it. Therefore, we have the following events.

- Event 3: A client  $j$  joins  $A$ . This event happens exactly once for each client. For each client  $j \notin A$ , it happens when  $\theta = \alpha_j$ .
- Event 4: A client  $j \in A$  is removed from  $A$ . By the description for the algorithm it happens in one of following five situations, and they can happen at most once for each client (so in total  $n$  times):

- $j$  gets a witness  $i$ : This can happen if  $j$  gets an edge to an already tight facility  $i$ , or  $i$  becomes tight with  $B(t_i) \leq B(\alpha_j)$ . In Event 1 we have discussed the situation in which a client gets a tight edge to a facility, therefore here we only focus on the time that facility  $i$  becoming tight. Note that the facility  $i$  might remove  $j$  from  $A$  only if  $B(t_i) \leq B(\alpha_j)$ , so we only consider such facilities in this case. Notice that these facilities are easy to identify because, when considering this event, we assume that no client or  $\theta$  change bucket (those events are considered separately) and therefore  $B(t_i)$  and  $B(\alpha_j)$  stay constant. A facility  $i$  can only become tight if there is no client in its neighborhood that is decreasing, so we also restrict ourselves to such facilities (which are again easy to identify).

The time  $\theta$ , that  $i$  becomes tight can now be calculated by solving the following equation:

$$z_i = \sum_{j \in \mathcal{D} \setminus A} [\alpha_j - d(i, j)^2]^+ + \sum_{j \in A} [\theta - d(i, j)^2]^+.$$

- $j$  is stopped by some client  $j'$ . This can only happen if  $j \in A$  and  $j' \notin A$ . The time  $\theta$  that this event happens can be computed by solving the equation  $2\sqrt{\theta} = d(j, j') + 6\bar{\alpha}_{j'}$ .
- For a client  $j \in U$ ,  $\alpha_j$  increases by  $\epsilon_z$ . The time  $\theta$  that this event happens is  $\theta = \alpha_j^{(1)} + \epsilon_z$  where  $\alpha_j^{(1)}$  denotes the  $\alpha$ -value of  $j$  at the beginning of the present call to SWEEP.
- $\alpha_j \geq \alpha_j^{(0)}$  and  $\alpha_j \geq \theta_s$ . This happens at the maximum time that each of the inequalities becomes tight. That is, at the time  $\theta = \max(\alpha_j^{(0)}, \theta_s)$ .
- There is a client  $j'$  that has already been removed from  $A$  such that  $\bar{\alpha}_j \geq d(j, j') + \bar{\alpha}_{j'}$ . This case is similar to the case that  $j$  is stopped by  $j'$ . That is, the time it happens can be calculated by solving the equation  $\sqrt{\theta} = d(j, j') + \bar{\alpha}_{j'}$ .

Now we focus on (3), i.e., the set of potentially tight facilities. In the following we explore the events that make facilities become potentially tight.

- Event 5: A facility  $i$  becomes potentially tight. This can only happen if one of the following situations occur:
  - A new client  $j$  joins  $N(i)$ : This is equivalent to saying that the edge  $(i, j)$  becomes tight. We have already considered this case.
  - $\alpha_j$  becomes more than  $\alpha_j^{(0)}$ : This can happen only if  $\alpha_j$  is increasing. Moreover, since  $\alpha_j^{(0)}$  is fixed throughout the call to SWEEP, this might happen at most once for each client (using that if an  $\alpha$ -value increases, it will not decrease later on). For any such  $j$ , time  $\theta$  that this event happens is easy to compute:  $\theta = \alpha_j^{(0)}$ .
  - For all  $j \in N^{(0)}(i)$ ,  $\alpha_j \geq \alpha_j^{(0)}$ : We can compute for each  $j \in N^{(0)}(i)$  the time that this inequality becomes tight, i.e.,  $\theta = \alpha_j^{(0)}$ . This event can occur only if  $\alpha_j$  is increasing for some  $j \in N^{(0)}(i)$  with  $\alpha_j < \alpha_j^{(0)}$ , and in this case it may happen at time  $\theta = \alpha_j^{(0)}$ . Again, this event might happen at most  $n$  times, since no  $\alpha$ -value is decreased after being increased.
- Event 6: A facility is no longer potentially tight. This event is similar to the previous one, except we consider the clients that are decreasing.

Finally we focus on (4) and (5) in Event 7 and Event 8, respectively.

- Event 7: A client  $j$  becomes the maximum client connected to a facility  $i \in P$ . We are interested in this case since it might result in decreasing  $\alpha_j$ . We decrease  $\alpha_j$  only if  $B(\theta) < B(\alpha_j)$ , so we do not need to consider this event if  $j \in A$ . For the remaining clients, this event can only happen if either the previous maximum client  $j'$  loses its tight edge to  $i$  or  $\alpha_j = \alpha_{j'}$ . We have already considered the case that tight edges change, so we focus on the time that  $\alpha_j = \alpha_{j'}$ . The time  $\theta$  that the equality happens can be computed as

$$\alpha_j = -|A|(\theta^{(0)} - \theta) + \alpha_{j'}.$$

Notice that above we assume that  $\alpha_j$  is not already decreasing. This is without loss of generality, since if  $\alpha_j$  is already decreasing then it cannot become equal to  $\alpha_{j'}$ , as  $\alpha_{j'}$  is decreasing by at most the same rate as  $\alpha_j$  (namely,  $-|A|$ ).

This event can happen at most once for each client-facility pair assuming that the set  $P$  of potentially tight facilities and the set  $A$  of active clients do not change. The reason is as follows. If  $\alpha_j > \theta$  becomes the maximum  $\alpha$ -value of all clients in  $N(i)$  for some potentially tight facility  $i$ , it was not previously decreasing. Moreover, while  $\alpha_j$  does not decrease, it remains the maximum  $\alpha$ -value of all clients in  $N(i)$ . On the other hand, if it starts decreasing then it will not stop decreasing until at least one of the sets  $A$  or  $P$  has changed. In this case  $j$  will not become the maximum  $\alpha$ -value for any facility (for which it is not already the maximum). Therefore, as  $P$  and  $A$  both change polynomially many times, this event can also happen at most polynomially many times.

- Event 8: A client enters the bucket of  $\theta$  or  $\theta$  changes bucket. We can compute the times of these events exactly as discussed in the analysis of the running time of QUASISWEEP. The number of occurrences of these events is the same as before and the time that they happen can also be computed similarly.

The above shows that we can calculate the next event in polynomial time and that there are in total at most polynomially many events. It follows that SWEEP can be implemented to run in time that is polynomial in the number of clients  $n$  and facilities  $m$ .

## B Bounding the Distances

**Lemma B.1.** *By losing a factor  $(1 + 100/n^2)$  in the approximation guarantee, we can assume that the squared-distance between any client and any facility is in  $[1, n^6]$ , where  $n = |\mathcal{D}|$ .*

*Proof.* We prove that for a given instance of the  $k$ -means problem,  $\mathcal{I} = (\mathcal{F}, \mathcal{D}, d, k)$ , we can in polynomial time output an instance  $\mathcal{I}' = (\mathcal{F}, \mathcal{D}, d', k)$  such that:

- The squared distance between any client and any facility is in  $[1, n^6]$  in  $\mathcal{I}'$ , i.e., for any  $i \in \mathcal{F}, j \in \mathcal{D}$ , we have  $1 \leq d'(i, j)^2 \leq n^6$ .
- For any constant  $\rho$ , any  $\rho$ -approximate solution for  $\mathcal{I}'$  is a  $\rho(1 + 100/n^2)$ -approximate solution for  $\mathcal{I}$ .

In what follows, we first prove the lemma for the case that  $d$  is a metric distance function, then we prove it for the case that  $d$  is a Euclidean metric function.



**Metric Distance:** We focus on the case that  $d$  is a metric distance. To that end, we create 3 instances  $\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}'$  with distances  $d_1, d_2, d'$  respectively. Choose  $M$ , such that  $\text{OPT}(\mathcal{I}) \leq M \leq 100\text{OPT}(\mathcal{I})$ . We can use the algorithm presented in [14] to find such  $M$ . First, let  $d_1(i, j) = \sqrt{\frac{n^3}{M}}d(i, j)$  for all  $i \in \mathcal{F}, j \in \mathcal{D}$ . This results in  $\text{OPT}(\mathcal{I}_1) = \text{OPT}(\mathcal{I})\frac{n^3}{M}$ , so  $n^3/100 \leq \text{OPT}(\mathcal{I}_1) \leq n^3$ . Second, for any  $i \in \mathcal{F}, j \in \mathcal{D}$  let  $d_2(i, j) = \min(d_1(i, j), n^2)$ . Consider any constant  $\rho$ -approximate solution, for  $\mathcal{I}_1$ . This solution cannot use any of the edges that we updated in the previous step, since the cost of this edge is more than  $n^4 \geq n\text{OPT}(\mathcal{I}_1)$ . Similarly, any  $\rho$ -approximate solution for  $\mathcal{I}_2$  cannot use any such edge. Therefore,  $\text{OPT}(\mathcal{I}_2) = \text{OPT}(\mathcal{I}_1)$ . Third, for any  $i \in \mathcal{F}, j \in \mathcal{D}$ , assign  $d'(i, j) = \max(d_2(i, j), 1)$ . Since this step might increase the cost of any solution by at most  $n$ ,  $\text{OPT}(\mathcal{I}_2) \leq \text{OPT}(\mathcal{I}') \leq \text{OPT}(\mathcal{I}_2) + n$ . Now it is clear that for any  $i \in \mathcal{F}, j \in \mathcal{D}$ ,  $1 \leq d'(i, j)^2 \leq n^4$ . We need to show that any good solution for  $\mathcal{I}'$  is also a good solution for  $\mathcal{I}$ . Note that during all these steps, we focused on the distances between clients and facilities. For guaranteeing that the  $d'$  is metric, we make the exact same changes on the pairs of facilities and pairs of clients as well.

Consider a  $\rho$ -approximate solution for  $\mathcal{I}'$ . We know that the cost of this solution is at most  $\rho \cdot \text{OPT}(\mathcal{I}')$ . Now consider the same solution for  $\mathcal{I}_2$ . Since the cost of any solution for  $\mathcal{I}_2$  is no more than its cost for  $\mathcal{I}'$ , the cost of this solution for  $\mathcal{I}_2$  is at most  $\rho \cdot \text{OPT}(\mathcal{I}') \leq \rho \cdot (\text{OPT}(\mathcal{I}_2) + n)$ . Also the cost of the same solution for  $\mathcal{I}_1$  equals to its cost for  $\mathcal{I}_2$  so it is at most  $\rho(\text{OPT}(\mathcal{I}_2) + n) = \rho(\text{OPT}(\mathcal{I}_1) + n) \leq \rho \cdot \text{OPT}(\mathcal{I}_1)(1 + 100/n^2)$ , where the last inequality is due to the fact that  $n^3/100 \leq \text{OPT}(\mathcal{I}_1)$ . Thus the cost of the same solution for  $\mathcal{I}$  is at most  $\frac{M}{n^3}(\rho \cdot \text{OPT}(\mathcal{I}_1)(1 + 100/n^2)) = \rho(1 + 100/n^2) \cdot \text{OPT}(\mathcal{I})$ . The lemma then follows by noting that  $d'$  is metric since we only rescaled, increased the minimum distance, and decreased the maximum distance of the given metric  $d$ .

**Euclidean Metric Distance:** Now assume that the given distance function is Euclidean. We assume that clients and facilities are points in some  $\ell$  dimensional Euclidean space. We first create a solution  $\mathcal{I}_1$ , making sure that the  $\text{OPT}(\mathcal{I}_1)$  is bounded by a polynomial. Similarly to before, we divide each coordinate by  $\sqrt{\frac{n^3}{M}}$ .<sup>9</sup> We get that  $d_1(i, j) = \sqrt{\frac{n^3}{M}}d(i, j)$  for all  $i \in \mathcal{F}, j \in \mathcal{D}$  and  $n^3/100 \leq \text{OPT}(\mathcal{I}_1) \leq n^3$ . Now we cluster the points in  $\mathcal{D} \cup \mathcal{F}$  such that the distance between any two points in different clusters is more than  $\Omega(n) \cdot \text{OPT}(\mathcal{I}_1)$ . To do that, we create each cluster as follows: Pick any client  $j$  that is not part of any cluster and add it to cluster  $S$ ; we call  $j$  the center of cluster  $S$ . While there exists a client  $j' \in \mathcal{D}$  that is not part of any cluster and the distance between  $j'$  to its closest client in  $S$  is less than  $n^2/4$  add  $j'$  to  $S$ . Let  $S_1, \dots, S_s$  be the clusters that we create. This gives a partition of the clients. Now we add a facility  $i$  to cluster  $S_\ell$ , if there exists a client  $j \in S_\ell$ , such that  $d(i, j) < n^2/8$ . This ensures that each facility is at most part of one cluster, since the distance between two clients in different clusters is more than  $n^2/4$ .

It is easy to see that our clusters have the following properties.

1.  $d_1(j, j')^2 < n^6/16$  for any two clients  $j, j'$  in the same cluster. The reason is, any client that we add to a cluster, the maximum distance in the cluster increases by less than  $n^2/4$  so  $d_1(j, j') < n^3/4$ .
2.  $d_1(i, j)^2 \leq n^6/8$  for any client  $j$  and facility  $i$  in the same cluster. The reason is, by the triangle inequality we have that  $d_1(i, j) \leq d_1(i, j_1) + d_1(j_1, j)$  where  $j_1$  is the client so that  $d_1(i, j_1) < n^2/4$ . we also know that  $d_1(j_1, j) < n^3/4$  by the previous property.

---

<sup>9</sup>We can still use [14] to find  $M$

3.  $d_1(i, i')^2 \leq n^6/8$  for any two facilities  $i, i'$  in the same cluster. Similarly to the previous case, let  $j, j'$  be the closest client in this cluster to  $i, i'$  respectively. By the triangle inequality we have that  $d_1(i, i') \leq d_1(i, j) + d_1(j, j') + d_1(j', i') \leq n^2/4 + n^3/4 + n^2/4$ .
4.  $d_1(i, j)^2 \geq n^4/64 \geq (n/64) \cdot \text{OPT}(\mathcal{I}_1)$  for any facility  $i$  and client  $j$  not in the same cluster.

We remove all the facilities that are not part of any cluster, since no client can be connected to them in any solution with approximation guarantee better than  $n/64$  (this follows from the above property 4). From above properties 1, 2, and 3 it is clear that the squared-distance between any two points in the same cluster is at most  $n^6/8$ . Now we continue by moving the center of the clusters to the origin and shift the whole cluster with them. Consider any two points  $p_1, p_2$  in the clusters and let  $j_1, j_2$  be the centers of those clusters respectively. We know that  $d_1(p_1, p_2) \leq d_1(p_1, j_1) + d_1(j_1, j_2) + d_1(j_2, p_2)$ . Also we know that  $j_1$  and  $j_2$  are both on the origin so  $d_1(p_1, p_2) \leq d_1(p_1, j_1) + d_1(j_2, p_2)$  and  $d_1(p_1, p_2)^2 \leq 2(d_1(p_1, j_1)^2 + d_1(j_2, p_2)^2)$ . Therefore,  $d_1(p_1, p_2)^2 \leq n^6/2$ . We add  $s$  new dimensions, one for each cluster. For each  $1 \leq i \leq s$ , we assign  $n^2$  to the  $i^{\text{th}}$  new coordinate for the points in  $i^{\text{th}}$  cluster and 0 to the rest. Let  $\ell' = \ell + s$  the number of the coordinates that the points have right now. Now consider two points and the value of their coordinates,  $j = (j_1, j_2, \dots, j_{\ell'})$ ,  $j' = (j'_1, j'_2, \dots, j'_{\ell'})$ , we know that

$$d(j, j')^2 = \sum_{k=1}^{\ell'} (j_k - j'_k)^2 = \sum_{k=1}^{\ell} (j_k - j'_k)^2 + \sum_{k=\ell+1}^{\ell'} (j_k - j'_k)^2 \leq n^6/2 + 2n^4.$$

Also, it still holds that any solution with an approximation guarantee better than  $n/64$  can only connect the clients in a cluster to the facilities in the same cluster, since the squared distance between any facility and any client in different clusters is at least  $n^4/64$ , which is more than  $n/64 \cdot \text{OPT}(\mathcal{I}_1)$ . Now we need to make sure that the distance between the facilities and the clients is at least one. To that end, we add one new dimension and assign one for facilities and zero for clients in this coordinate. Similarly to the analysis of the general metric, we can show that any  $\rho$ -approximate solution for the new instance is also a  $\rho(1 + 100/n^2)$ -approximate for  $\mathcal{I}$ , since we increase the cost of any solution by at most  $n$ . Note that the last step does not increase the distance-squared between any two points by more than one so the maximum distance-squared between any two points is at most  $n^6/2 + 2n^4 + 1 \leq n^6$ .

Clearly, the running time of this procedure is  $\text{poly}(n)$ .  $\square$

## References

- [1] A. Archer, R. Rajagopalan, and D. B. Shmoys. Lagrangian relaxation for the k-median problem: New insights and continuity properties. In *Proc. 11th ESA*, pages 31–42, 2003.
- [2] D. Arthur and S. Vassilvitskii. How slow is the k-means method? In *Proc. 22nd SoCG*, pages 144–153, 2006.
- [3] D. Arthur and S. Vassilvitskii. K-means++: The advantages of careful seeding. In *Proc. 18th SODA*, pages 1027–1035, 2007.
- [4] V. Arya, N. Garg, R. Khandekar, A. Meyerson, K. Munagala, and V. Pandit. Local search heuristics for  $k$ -median and facility location problems. *SIAM J. Comput.*, 33(3):544–562, 2004.
- [5] P. Awasthi, A. Blum, and O. Sheffet. Stability yields a PTAS for k-median and k-means clustering. In *Proc. 51st FOCS*, pages 309–318, 2010.

- [6] P. Awasthi, M. Charikar, R. Krishnaswamy, and A. K. Sinop. The hardness of approximation of euclidean k-means. In *Proc. 31st SoCG*, pages 754–767, 2015.
- [7] J. Byrka and K. Aardal. An optimal bifactor approximation algorithm for the metric uncapacitated facility location problem. *SIAM J. Comput.*, 39(6):2212–2231, 2010.
- [8] J. Byrka, T. Pensyl, B. Rybicki, A. Srinivasan, and K. Trinh. An improved approximation for  $k$ -median, and positive correlation in budgeted optimization. In *Proc. 26th SODA*, pages 737–756, 2015.
- [9] M. Charikar and S. Guha. Improved combinatorial algorithms for facility location problems. *SIAM J. Comput.*, 34(4):803–824, 2005.
- [10] F. A. Chudak and D. B. Shmoys. Improved approximation algorithms for the uncapacitated facility location problem. *SIAM J. Comput.*, 33(1):1–25, 2004.
- [11] V. Cohen-Addad, P. N. Klein, and C. Mathieu. The power of local search for clustering. *CoRR*, abs/1603.09535, 2016.
- [12] D. Feldman, M. Monemizadeh, and C. Sohler. A PTAS for k-means clustering based on weak coresets. In J. Erickson, editor, *Proc. 23rd SoCG*, pages 11–18, 2007.
- [13] Z. Friggstad, M. Rezapour, and M. R. Salavatipour. Local search yields a PTAS for k-means in doubling metrics. *CoRR*, abs/1603.08976, 2016.
- [14] A. Gupta and K. Tangwongsan. Simpler analyses of local search algorithms for facility location. *CoRR*, abs/0809.2554, 2008.
- [15] A. K. Jain. Data clustering: 50 years beyond k-means. *Pattern Recogn. Lett.*, 31(8):651–666, June 2010.
- [16] K. Jain, M. Mahdian, E. Markakis, A. Saberi, and V. V. Vazirani. Greedy facility location algorithms analyzed using dual fitting with factor-revealing LP. *J. ACM*, 50:795–824, 2003.
- [17] K. Jain, M. Mahdian, and A. Saberi. A new greedy approach for facility location problems. In *Proc. 34th STOC*, pages 731–740, 2002.
- [18] K. Jain and V. V. Vazirani. Approximation algorithms for metric facility location and  $k$ -median problems using the primal-dual schema and lagrangian relaxation. *J. ACM*, 48(2):274–296, 2001.
- [19] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu. A local search approximation algorithm for k-means clustering. *Comput. Geom.*, 28(2-3):89–112, 2004.
- [20] E. Lee, M. Schmidt, and J. Wright. Improved and simplified inapproximability for k-means. *CoRR*, abs/1509.00916, 2015.
- [21] S. Li. A 1.488 approximation algorithm for the uncapacitated facility location problem. *Inf. Comput.*, 222:45–58, 2013.
- [22] S. Li and O. Svensson. Approximating k-median via pseudo-approximation. *SIAM J. Comput.*, 45(2):530–547, 2016.

- [23] J. Lin and J. S. Vitter. Approximation algorithms for geometric median problems. *Inf. Process. Lett.*, 44:245–249, 1992.
- [24] S. Lloyd. Least squares quantization in PCM. *IEEE Trans. Inf. Theor.*, 28(2):129–137, Sept. 2006.
- [25] J. Matoušek. On approximate geometric k-clustering. *Discrete & Computational Geometry*, 24(1):61–84, 2000.
- [26] R. Ostrovsky, Y. Rabani, L. J. Schulman, and C. Swamy. The effectiveness of Lloyd-type methods for the k-means problem. *J. ACM*, 59(6):28:1–28:22, Jan. 2013.
- [27] D. B. Shmoys, E. Tardos, and K. Aardal. Approximation algorithms for facility location problems (extended abstract). In *Proc. 29th STOC*, pages 265–274, 1997.
- [28] A. Vattani. k-means requires exponentially many iterations even in the plane. *Discrete & Computational Geometry*, 45(4):596–616, 2011.